



A RELIABILITY STUDY OF A TELEOPERATED ROBOTIC
SYSTEM WITH APPLICATION TO THE NEXT
GENERATION MUNITIONS HANDLING SYSTEM

THESIS

Matthew J. Rummer
Second Lieutenant, USAF

AFIT/GOR/ENY/97M-02

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DTIC QUALITY INSPECTED 1

AFIT/GOR/ENY/97M-02

A RELIABILITY STUDY OF A TELEOPERATED ROBOTIC
SYSTEM WITH APPLICATION TO THE NEXT
GENERATION MUNITIONS HANDLING SYSTEM

THESIS

Matthew J. Rummer
Second Lieutenant, USAF

AFIT/GOR/ENY/97M-02

Approved for public release; distribution unlimited

19970402 088

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government

AFIT/GOR/ENY/97M-02

A RELIABILITY STUDY OF A TELEOPERATED ROBOTIC SYSTEM
WITH APPLICATION TO THE
NEXT GENERATION MUNITIONS HANDLING SYSTEM

THESIS

Presented to the Faculty of the Graduate School of Engineering
of the Air Force Institute of Technology
Air University

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Operations Research

Matthew J. Rummer, B.S.M.E.
Second Lieutenant, USAF

March 1997

Approved for public release; distribution unlimited

THESIS APPROVAL

Student: Matthew J. Rummer, Second Lieutenant, USAF

Class: GOR-97M

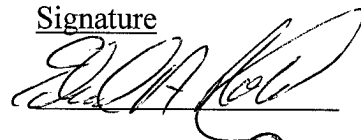
Title: A Reliability Study of a Teleoperated Robotic System with Application to the Next Generation Munitions Handling System

Defense Date: 28 February 1997


Committee: Name/Title/Department

Signature

Advisor Edward A. Pohl, Major, USAF
Assistant Professor
Department of Aeronautics and Astronautics



Reader Dean L. Schneider, Major, USAF
Assistant Dean for Academic Affairs
The Graduate School of Engineering



Reader William P. Murdock, Major, USAF
Assistant Professor
Department of Operational Sciences



Acknowledgments

I would like to thank my thesis advisor, Major Ed Pohl, for his help and guidance throughout this project. Also, special thanks to Major Dean Schneider for his interest and eagerness to contribute to this project. I would like to thank the Robotics Process and Systems Division at Oak Ridge National Laboratory for their willingness to assist with this research project. Their innovative and professional work will undoubtedly benefit the United States Air Force. I would also like to thank Dr. D. W. Repperger for his expertise concerning the human reliability field. Finally, a special thank you to my wife, Stefanie, for her patience and understanding throughout this research effort.

Matthew J. Rummer

Table of Contents

	Page
Acknowledgments.....	iii
List of Figures	viii
List of Tables	x
Abstract.....	xii
Chapter 1. Introduction	1-1
1.1 Background	1-1
1.1.1 Robotics and Reliability	1-1
1.1.2 Designing for Fault-Tolerance.....	1-3
1.2 Application.....	1-4
1.3 Problem Statement and Objectives	1-4
1.4 Outline.....	1-6
Chapter 2. Literature Review	2-1
2.1 Introduction.....	2-1
2.2 General Reliability Concepts	2-1
2.3 Hardware Reliability	2-3
2.4 Kinematic Reliability	2-5
2.5 Software Reliability	2-7
2.6 Human Reliability	2-11
2.6.1 Human Error Data Banks	2-12
2.6.2 Human Reliability Models	2-13

2.6.3 Human Error Estimation Techniques	2-14
2.6.4 Human Error Recovery	2-17
2.7 Combinations of Reliability Models	2-18
2.7.1 Human-Machine Models	2-19
2.7.2 Robotics	2-20
2.8 Summary	2-21
Chapter 3. Model Development	3-1
3.1 Introduction	3-1
3.1.1 General Telerobotic System Description	3-2
3.2 First-Order Markov Model	3-4
3.2.1 Example of First-Order Markov Model	3-6
3.2.2 Results from First-Order Markov Model Example	3-8
3.3 Expanded Markov Reliability Model	3-11
3.3.1 Example of Expanded Markov Reliability Model	3-12
3.3.2 Results from Expanded Markov Model Example	3-21
3.4 Expanded Semi-Markov Model	3-23
3.5 Reliability Index Development	3-25
3.6 Summary	3-27
Chapter 4. Model Application	4-1
4.1 Introduction	4-1
4.2 Model Application	4-3
4.2.1 Human	4-4

4.2.2 Hardware	4-7
4.2.3 Software.....	4-12
4.2.4 Kinematic	4-14
4.3 Results.....	4-18
4.3.1 Task Analysis	4-19
4.3.2 Kinematic Analysis	4-21
4.4 Optimization Issues.....	4-28
4.5 Summary	4-29
Chapter 5. Conclusions and Recommendations.....	5-1
5.1 Summary of Research	5-1
5.2 Recommendations.....	5-2
5.2.1 Human Component.....	5-3
5.2.2 Kinematic Component.....	5-3
5.2.3 Hardware and Software Components	5-5
5.2.4 Fault Tolerance	5-5
5.2.5 Continuous Model	5-6
5.2.6 Trajectory Planning	5-6
5.3 Conclusion	5-7
Appendix A. First-Order Markov Model Solution	A-1
Appendix B. SLAM II Program and Output for ATD System	B-1
Appendix C. Plots of R_k Simulation results (d3 Locked)	C-1
Appendix D. Plots of R_k Simulation Results (θ_1 Locked).....	D-1

Appendix E. FORTRAN Program for R_k Calculations.....	E-1
Appendix F. Schematic of Joint Variables for ATD System.....	F-1
Appendix G. Pitch, Roll, and Yaw Angle Formulation.....	G-1
Appendix H. Trajectories for ATD System	H-1
Bibliography.....	BIB-1
VITA	VITA-1

List of Figures

Figure	Page
2.1 Bathtub Curve for Hazard Function	2-4
3.1 Schematic of Telerobotic System	3-2
3.2 First-Order Markov Model	3-5
3.3 Sensitivity Analysis on Probability of Recovery	3-10
3.4 Three Link Manipulator Used for Example Calculations	3-13
3.5 Straight Line Trajectory of Three Link Manipulator	3-20
3.6 Sensitivity Analysis on Probability of Recovery	3-23
4.1 Illustration of Advanced Technology Demonstrator for the NGMH System	4-2
4.2 Two-State Human Markov Model	4-5
4.3 Trajectories for Positioning of Manipulator	4-17
4.4 X-Y Coordinates over Positioning Workspace	4-22
4.5 Type III Kinematic Reliabilities over Positioning Workspace (3-D Plot)	4-23
4.6 Type III Kinematic Reliabilities over Positioning Workspace (Gray Scale)	4-24
4.7 Kinematic Reliability Values for Joint Configurations (Positioning)	4-25
4.8 Kinematic Reliability Values for Joint Configurations (Orientation)	4-26
A.1 SLAM II Network Diagram for First-Order Model	A-3
B.1 SLAM II Network Diagram for ATD System	B-3

C.1 Joint Variations within Kinematic ConfigurationsC-2 (d_3 varied)	C-2
C.2 R_k Values for Joint Configurations ($d_3=74.56\text{in}$).....C-2	C-2
C.3 R_k Values for Joint Configurations ($d_3=80.56\text{in}$).....C-3	C-3
C.4 R_k Values for Joint Configurations ($d_3=86.56\text{in}$).....C-3	C-3
C.5 R_k Values for Joint Configurations ($d_3=92.56\text{in}$).....C-4	C-4
C.6 R_k Values for Joint Configurations ($d_3=98.56\text{in}$).....C-4	C-4
D.1 Joint Variations within Kinematic Configurations D-1 (θ_1 varied)	D-1
D.2 R_k Values for Joint Configurations ($\theta_1=40^\circ$)..... D-2	D-2
D.3 R_k Values for Joint Configurations ($\theta_1=54^\circ$)..... D-2	D-2
D.4 R_k Values for Joint Configurations ($\theta_1=68^\circ$)..... D-3	D-3
D.5 R_k Values for Joint Configurations ($\theta_1=82^\circ$)..... D-3	D-3
D.6 R_k Values for Joint Configurations ($\theta_1=96^\circ$)..... D-4	D-4
D.7 R_k Values for Joint Configurations ($\theta_1=110^\circ$)..... D-4	D-4
F.1 Schematic for First Four Joints..... F-1	F-1
F.2 Schematic for Last Four Joints F-2	F-2
G.1 Pitch, Roll, and Yaw Angles..... G-2	G-2

List of Tables

Table	Page
3.1 Variables of First-Order Markov Model	3-6
3.2 Values used in First-Order Markov Model Example	3-7
3.3 Human Error Probabilities (HEP) for Each Subtask	3-14
3.4 Example Software Module Characteristics	3-15
3.5 Example Software Control Transfer Frequencies	3-15
3.6 Hardware Failure Rate Values.....	3-17
3.7 Positions Along Straight Line Trajectory	3-20
3.8 Kinematic Failure Rate Estimates	3-20
3.9 Summary Table for Reliability After 50 Repetitions	3-21
3.10 Summary of Simulation Results.....	3-22
3.11 Summary of Semi-Markov Simulation Results.....	3-24
3.12 Summary of Semi-Markov Simulation Results..... (without Kinematic Component)	3-26
3.13 Reliability Index Calculations	3-27
4.1 Scale Factors and Human Failure Rates	4-6
4.2 Summary of Hydraulic Components Selected.....	4-8
4.3 Summary of Electrical Components Selected	4-9
4.4 Summary of Bearing Life Calculations	4-10
4.5 Summary of Hardware Components	4-11
4.6 Control Resolution for Each Joint.....	4-14

4.7 Orientation Used for All Trajectories.....	4-17
4.8 Minimum Kinematic Reliability Values	4-18
4.9 Parameters for Semi-Markov Simulation.....	4-19
4.10 Summary of Semi-Markov Simulation Results.....	4-19
4.11 Reliability Index Values	4-20
H.1 Trajectory 1 Positioning	H-1
H.2 Trajectory 1 Orientation	H-1
H.3 Trajectory 2 Positioning	H-2
H.4 Trajectory 2 Orientation	H-2
H.5 Trajectory 3 Positioning	H-2
H.6 Trajectory 3 Orientation	H-2

Abstract

The United States Air Force is currently developing an Advanced Technology Demonstrator (ATD) for the Next Generation Munitions Handling System (NGMH). The NGMH/ATD system is being designed at Oak Ridge National Laboratory as a teleoperated robotic manipulator. In this research, a reliability model for a general telerobotic manipulator is developed including: (1) human, (2) software, (3) hardware, and (4) kinematic components. Each of these components contributes towards the reliability of the system, where reliability is defined as the probability of the end-effector of the manipulator being within a certain error bound. The model provides a comparative measure of reliability based on a combination of two underlying models. A semi-Markov model is used for the human, software, and hardware components, while the kinematic reliability estimates are obtained through a separate simulation. The reliability model developed is not intended to be a predictive model, rather to provide a means of comparing different configurations of a telerobotic system.

A RELIABILITY STUDY OF A TELEOPERATED ROBOTIC SYSTEM
WITH APPLICATION TO THE
NEXT GENERATION MUNITIONS HANDLING SYSTEM

Chapter 1: Introduction

1.1 Background

Reliability is an important aspect of engineering that should be carefully considered during the design of any system. Human nature tends to cause engineers to focus only on the success of their designs which may neglect considering even the possibility of failure. However, failures do and will continue to occur in even the most sophisticated designs. Often the degree to which reliability is considered depends on the consequences of a system failure. Because of this, it makes sense that a high degree of reliability emphasis has been used in the nuclear power industry as well as the space industry. A considerable amount of reliability research has gone into both of these fields. However, the benefits of a reliable system are encouraging the spread of reliability based design into many different fields. One such field is robotics.

1.1.1 Robotics and Reliability

The reliability of a system is generally defined as the probability that the system will perform its intended function for a specified interval of time under stated conditions [10:1]. Before this definition can be applied to any real world system, the function of the

system, the conditions of operation, the time interval, and most importantly, the failure of the system must be clearly defined. Systems will obviously have different measures of failure based not only on the task of the system but on the impact that a failure will have. The reliability of a robotic system can be defined as the probability of the end effector being within a certain error bound in both its position and orientation [3]. Thus for a robotic system, a failure can be defined as the end effector being outside of this error bound.

In robotic systems, reliability has become an increasingly important issue. Unfortunately, reliability is often difficult to predict due to the complex nature of many robotic systems. A number of reliability studies within the field of robotics have attempted to model, predict, and/or improve reliability within robotic systems. These studies have varied from a cursory estimation of robot reliability to a very thorough analysis of the entire robot system [29]. Most studies are directed towards specific systems with little attempt at achieving generality. The issue of safety within robotic systems has played a key role in many of these studies. The research mentioned here has focused primarily on the fully automated robotic system.

A robotic system can be operated autonomously (pre-programmed) or can be teleoperated (remote-controlled). While the teleoperated robotic system has a human in the control loop of the manipulator, some of the manipulator's functions may still be autonomous, i.e. a combination of automated and remote-controlled inputs. The degree of human interaction is dependent upon the task, the amount of responsibility given to the operator, the need for decision making during operation, the sophistication of software

etc. In any case, a teleoperated system introduces the reliability of the human operator into the overall system reliability. The problem of including the human operator in a telerobotic system presents both a challenge and opportunity for improved reliability within the system. By allowing the operator to detect faults within the system, and having software that can correct for operator and system faults as well, the overall reliability of a teleoperated system should be improved.

1.1.2 Designing for Fault-Tolerance

A field closely related to reliability engineering is that of designing for fault-tolerance. Fault-tolerance is the ability of a system to recover from the effects of faults before complete failure occurs. Fault-tolerance can be incorporated into almost any system with a variety of techniques. Redundancy within the system is a common form of introducing fault-tolerance into the design. In robotic systems, redundancy is often accomplished by adding an extra degree of freedom to the manipulator. With a teleoperated manipulator, fault-tolerance could also be achieved by allowing the operator to perform fault detection and make corrections or adjustments to the system before a complete failure occurs. Similarly, the system software could recognize and correct for possible human error. This human/system interaction should serve to reinforce system reliability and fault-tolerance within the system. Fault-tolerance should be carefully employed in order to improve the reliability of a system.

1.2 Application

The United States Air Force is currently developing the Next Generation Munitions Handler (NGMH) in cooperation with the Navy and Marine Corps. This munitions handling system is being designed as a teleoperated robotic manipulator. An Advanced Technology Demonstrator (ATD) for this system is currently being developed at Oak Ridge National Laboratory to demonstrate the applicability of this new technology. One of the main goals during the ATD development is to design the manipulator with optimal mechanical and human interface configurations. Obviously, a munitions handling system should be highly reliable, not only for safety reasons but also for efficiency on the flight line. This research effort is being undertaken in order to contribute to the understanding of reliability within the NGMH/ATD system.

1.3 Problem Statement and Objectives

Previous research into the area of robotic reliability has focused mainly on fully automated robot systems. Reliability models currently exist for robot systems which include hardware, kinematic, and software components [29]. This thesis effort attempts to further analyze the reliability of a teleoperated robotic system by including the human operator in the model as well. The reliability components of a telerobotic system can be divided into four main categories: (1) human, (2) software, (3) hardware, and (4) kinematic reliability. A complete reliability model of a teleoperated manipulator should include all of these components.

The goal of this thesis effort is to develop a mathematical model of the teleoperation system reliability that will include human, software, hardware, and

kinematic components within the system model. For the purposes of this project, the system reliability will be defined as the probability of the teleoperator's end-effector being within a certain error, ϵ , designated as $P(S_{TR})$. A preliminary model of this probability is given below:

$$P(S_{TR}) = P(S_{hw})P(S_{kin})[P(S_{human}|S_{sw})P(S_{sw}) + P(S_{sw}|S_{human})P(S_{human})] \quad (1.1)$$

where: $P(S_{hw})$ is the probability of success of the hardware,

$P(S_{kin})$ is the probability of success of the kinematic configuration,

$P(S_{human})$ is the probability of success of the human operator, and

$P(S_{sw})$ is the probability of success of the software.

This preliminary model serves as a starting point for developing a more complete mathematical model of the telerobotic system reliability.

In order to further develop this model, the four main components within the system will be analyzed individually to determine an appropriate reliability model for each. These four reliability models will then be combined to form an overall reliability model for the entire system. A major factor in selecting the component models will be their compatibility with other models. The models for kinematic and human reliability will be of particular interest, since these areas are often overlooked in robotic reliability.

Once a reliability model for a telerobotic system has been developed, this model will be applied to the NGMH/ATD system. The purpose of this model application is primarily to obtain insights concerning the reliability of the system. It is hopeful that these insights may be useful in the development of the system. Based on this model application, optimization issues within the NGMH/ATD system will also be presented.

The final phase of this project involves recommendations for future consideration. These recommendations include ideas for further refinement of model as well as practical uses of the model. A number of research opportunities are available as an extension to the work done here.

1.4 Outline

A brief description of the remainder of this report is as follows. Chapter 2 presents a literature review of the various components of the overall reliability model to be considered in this research. The main components of this model are hardware, kinematic, software, and human reliability. These components are looked at individually and current combinations of these models are also presented.

Chapter 3 describes the reliability model formulation for the teleoperated robotic system using the four main components given previously. The presentation of the model development begins with a simple first-order Markov model. This model is then expanded to represent each subtask within the overall task of a system. Next, a semi-Markov model is developed to allow non-constant failure rates within the system. Finally, a reliability index is introduced as a comparative measure of system reliability. Examples are given to illustrate the use of these models.

Chapter 4 describes the application of the reliability index model to the NGMH/ATD system. Each of the four main components of a telerobotic system are analyzed in the application of the model. Optimization issues within the NGMH/ATD system are also discussed.

Chapter 5 provides a summary of the entire research effort. The key results of this research are discussed and recommendations are given for future consideration.

The report also has a number of appendices attached. These additions provide details that were left out of the body of the report in order to make it more readable. Appendix A contains details related to the first-order Markov model example. Both an analytic and simulation solution are presented here. Appendix B contains details of the SLAM II program used for the semi-Markov solution for the ATD system. Appendix C and D contain plots of kinematic reliability estimates for various configurations of the ATD system. Appendix E provides the FORTRAN code used to generate kinematic reliability point estimates for the ATD system. Appendix F illustrates the joint variables for the ATD system. Appendix G provides the development of the roll, pitch, and yaw angles used to determine the orientation of the manipulator. And finally, Appendix H provides details of three arbitrary trajectories used for example calculations of the ATD system.

Chapter 2: Literature Review

2.1 Introduction

The purpose of this literature review is to introduce current reliability estimation techniques and models which will be considered for the overall model of the teleoperated robotic system. Current reliability models of the individual components to be incorporated into the teleoperated robotic system are presented. These components include: hardware, kinematic, software, and human reliability. Reliability models for systems with multiple components are also introduced. These system models include general man-machine studies as well as robotic applications.

2.2 General Reliability Concepts

This section is intended to provide a brief overview of basic reliability concepts. Recall that reliability has been defined as the probability that the system will perform its intended function for a specified interval of time under stated conditions [11:1]. From this definition it is apparent that the basic theories of probability and statistics are essential in the reliability field. The probability of failure as a function of time can be defined as

$$P(\mathbf{T} \leq t) = F(t), \quad t \geq 0 \quad (2.1)$$

where \mathbf{T} is a random variable denoting failure time, and $F(t)$ represents the probability that the system will fail by time t . The reliability function, $R(t)$, can now be expressed as

$$R(t) = 1 - F(t) = P(T > t) \quad (2.2)$$

If the time to failure random variable T has a density function $f(t)$, then

$$R(t) = 1 - F(t) = 1 - \int_0^t f(\tau) d\tau = \int_t^{\infty} f(\tau) d\tau \quad (2.3)$$

Another useful function in reliability is the hazard function, $h(t)$, which describes the instantaneous failure rate. The hazard function can be defined as

$$h(t) = \frac{f(t)}{1 - F(t)} = \frac{f(t)}{R(t)} \quad (2.4)$$

Using the hazard function, reliability can now be expressed as

$$R(t) = \exp \left[- \int_0^t h(\tau) d\tau \right] \quad (2.5)$$

An additional reliability measure, commonly referred to in the reliability field, is the mean time to failure (MTTF). In terms of system reliability, the MTTF can be represented as

$$MTTF = \int_0^{\infty} R(t) dt \quad (2.6)$$

In summary, four related reliability functions have been presented. These characteristic functions include: the cumulative distribution function, $F(t)$, the probability density function, $f(t)$, the reliability function, $R(t)$, and the hazard function, $h(t)$. Additionally, the MTTF of a system has been presented in terms of system reliability. These characteristic functions and concepts are the building blocks of any reliability study. For additional reading refer to [11,23].

2.3 Hardware Reliability

Hardware reliability is the most common component of any reliability analysis. The basic definition given for system reliability can easily be applied to hardware reliability. Using the basic definition presented earlier [11:1], hardware reliability is the probability that a hardware component will perform its intended function for a specified interval of time under stated conditions. The hardware components within a system can be both mechanical and electrical devices. Many databases have been created to assist in hardware reliability studies. These databases consist mainly of failure rate data for various system components. MIL-HDBK-217F is a source of failure rate data commonly used for electronic components [18]. A source of failure rate data for non-electronic components is the *Non-Electronic Parts Reliability Data* (NPRD) handbook [24]. Failure rate data from these and other databases can be used to model various components of a system.

Many reliability studies assume constant failure rates for hardware components. This is not always a good assumption due to the phenomenon commonly referred to as the bathtub curve. The bathtub curve is a typical shape of the hazard function plotted against time for populations of most hardware components [23:8]. Figure 2.1 depicts the shape of a generic bathtub curve. Notice that the bathtub curve can be divided into three sections. The early section of the curve is decreasing due to early failures, the middle section of the curve is fairly constant, and the last section of the curve is increasing due to wearout. The wearout portion of the curve could be related to a number of factors such as aging, fatigue, wear, corrosion, etc. Almost all hardware components exhibit some form

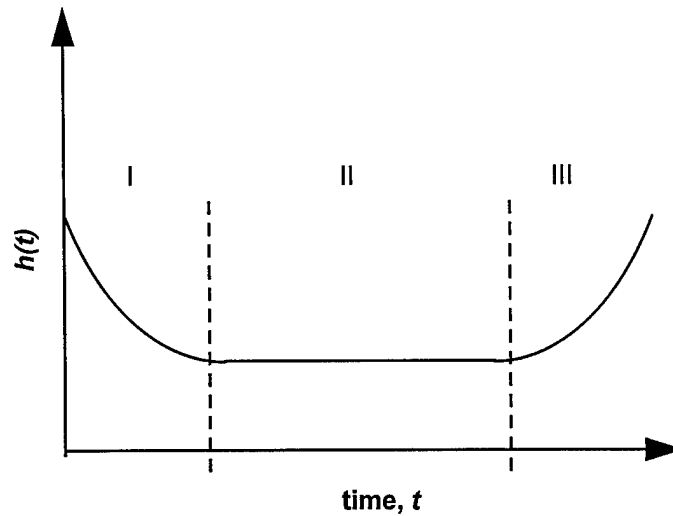


Figure 2.1 Bathtub Curve for Hazard Function

of deterioration towards the end of its useful life. Because of this bathtub curve effect, constant failure rate models are not always the best choice for hardware components.

Another method to analyze hardware components involves performing a statistical analysis of failure data. The first step in this process is to collect failure data for the particular hardware component(s) to be analyzed. Common failure distributions that are used to describe failure data include the Weibull, normal, lognormal, and exponential [23:Ch. 4]. Once a distribution has been selected, hardware reliability models can be derived for the various hardware components. These models can then be used to assist in determining system reliability.

A thorough discussion of hardware components commonly used in robotic systems has been presented in [29]. The hardware components described include cables, connectors, sensors, actuators, motors, gears, and end effectors. A description of each component is given along with a discussion of its relationship within a robotic system. A

summary of failure data is given where available. This summarized data includes failure rates and/or MTBF (mean time between failure) values for each type of component considered.

2.4 Kinematic Reliability

Kinematics, in general, is the study of motion without regard to forces or other factors that influence the motion [12:168]. Robot kinematics involves the study of the spatial configuration of the robot manipulator as a function of time without regard to the forces that cause the motion [3]. A robotic manipulator is composed of a number of links connected together by joints. The links and joints of a manipulator form a kinematic chain which is open at one end and connected to ground at the other. The end-effector of a robot manipulator is connected to the open end of this chain, and is generally some type of gripper or tool used to perform a task. Recall that the reliability of a robotic system has been defined as the probability of the end effector being within a certain error bound in both its position and orientation [3].

In order to specify geometric relationships between links within a robotic manipulator, four kinematic parameters are assigned to each link/joint. These parameters are used to completely specify distances and angles associated with each link as well as relative positions to other links. A common set of rules was established by Denavit and Hartenberg to be used in the representation of these parameters. Using these rules, the kinematic parameters can be used to create D-H matrices which can be used to relate link positions and orientations between coordinate frames. For a more complete description of robot kinematics and D-H representation refer to [12:Ch.8]. The relationship between

joint variables and the position and orientation of the end-effector is the primary focus of robot kinematics.

Bhatti and Rao presented a general method to perform kinematic reliability analysis of a robot manipulator in [3]. The main premise in their technique is to allow the kinematic parameters to be represented by random variables. Specifically, in their paper the kinematic parameters are all assumed to be normal random variables. The kinematic reliability of the manipulator is divided into positional reliability and orientational reliability. Two methods are presented to compute the kinematic reliability. An analytical method and a Monte-carlo simulation method are both used in numerical examples. Based on the results from these examples, the Monte-carlo simulation method was found to be the better method.

The kinematic reliability of a manipulator is dependent on the kinematic parameters associated with each link/joint. Because of this, the kinematic reliability will change as the configuration of the manipulator changes. Similarly, different end-effector locations will lead to different kinematic reliabilities. In order to use this technique, the end-effector location must be specified and reliability calculated at these locations. If more than one manipulator configuration is possible, then this technique can also be employed to select the one with the highest reliability.

An article by Paredis *et al* [22] discusses kinematic design of fault tolerant manipulators. They define a fault tolerant manipulator as an n degree of freedom manipulator that will still be able to meet task specifications, even if any one or more of its joints fail. A simple technique used to create fault tolerant manipulators is to design

every joint with a redundant actuator. This can lead to a costly overdesign of the system however. Their paper presents methodology that enables the design of a fault tolerant manipulator with the minimum amount of redundancy.

According to [3] there are three main sources of error leading to kinematic failure. The first source stems from errors in manufacture and assembly. Extremely high tolerances can be assigned to minimize these errors, however, that is often a costly alternative. The second source is from errors in the actuators and controllers of the links. Sensing devices can be used to provide feedback in an effort to decrease these errors. These errors cannot be totally eliminated due to the inherent limits of sensing devices. The third source of error is due to the deflection of the links. These errors are generally larger with high speed light weight manipulators. Friction at the joints of a manipulator also could lead to error in kinematics.

2.5 Software Reliability

Software reliability has been defined as the probability of failure-free operation of a computer program for a specified time in a specified environment [20:15]. Software reliability differs from the other reliability components in that the errors leading to a failure are embedded in the program from the start. While other components may have factors such as wearout or fatigue associated with their reliability, software errors are inherent in the program. Not all errors within the program will cause it to fail. Software reliability is concerned with the probability that existing errors will affect the output.

The principle causes of error in computer software stems from development faults. Three major sources of error are given in [21:236]. The first source is found in

the specification of the software. The objective of the software should be clearly specified to the programmers. Vague or ambiguous specifications can quickly lead to software error. The second source of error is the software system design. The software system design involves development of the program structure and logic to be used within the program. Incorrect interpretation of the specifications could lead to system design errors. The third source of error is the actual code generation. Because programs can involve a large amount of computer code, this source of error can be extremely problematical.

Two key elements of reliable software design are given in [21:238]. These elements are a structured program and modularity. A well structured program is less likely to have errors and is also easier to maintain. Modular programming allows the program to be represented by a group of smaller, reasonably sized programs. By allowing the program to be modular, the overall program can be easier to understand, and errors within the smaller programs can be identified and isolated.

The concept of fault tolerance can be incorporated into software design. Fault tolerance within the software component of the system is generally cheaper to introduce than fault tolerance of hardware components. One method of fault tolerance is to set up internal tests within the computer program to locate possible error sources. These errors can be located and corrected without causing the entire system to fail. Program redundancy is another means of fault tolerance. While redundant programs should perform the same task, they should not be identical. Identical programs will have the same internal faults and can result in the same failures. Different programming teams

may be used to generate redundant software to be used in the system. However, even with different programs, the same failure may occur if similar programming styles are used.

A number of software reliability models have been developed. Musa presents two primary models which are utilized throughout his book on software reliability [20]. The models given are the basic execution time model and the logarithmic Poisson execution time model. These models are further divided into execution time and calendar time formats. The initial modeling is done using execution time, or processor time used by the program. These models can then be converted to calendar time, which is more meaningful to most engineers. The execution time component for both models assumes that failures occur as a random process, specifically a nonhomogeneous Poisson process. Each model has a failure intensity function associated with it. These failure intensity functions are defined as instantaneous failure rates. Musa notes that if the program is operational, then the failure intensity can be assumed to be constant [20:49]. The software reliability in terms of execution time can now be written as

$$R(\tau) = \exp(-\lambda\tau) \quad (2.7)$$

where τ represents the execution time and
 λ is the constant failure intensity.

The execution time must be related to calendar time in order to calculate the calendar time component. Calendar time to execution time ratios are utilized in order to make these calculations. For further information regarding these calculations refer to [20:Ch. 2]. The choice of which model to select depends on a number of factors. Musa

recommends using the basic model for studies or predictions before execution of the software [20:59].

Other software reliability models are presented by Littlewood and Kubat [15, 13]. Both of these models of software reliability are based on a modular program structure. Within a modular program structure, the overall program is divided into separate modules and the control of the program can be transferred between the various modules. Each module within the program has a certain probability of error associated with it, and the transfer of control between modules may also lead to errors. The Littlewood model describes the failure rate of the overall program as

$$\lambda_{sw} = \sum_i a_i v_i + \sum_i b_{ij} \lambda_{ij} \quad (2.8)$$

where a_i is the proportion of time spent in module i ,
 b_{ij} is the frequency of transfer of control from module i to module j ,
 v_i is the failure rate for module i , and
 λ_{ij} is the probability of failure during transfer from i to j .

The Littlewood model is a fairly simple model that can be applied to almost any modular program. During the development and testing phase of software, data can be collected and applied to the Littlewood model to predict software reliability.

The Kubat software reliability model is an extension of the model presented by Littlewood. Kubat's model includes a representation of the amount of time each module spends in control. The Kubat model is a more complicated model of software reliability. The mathematics involved to determine software reliability are more complex than in the Littlewood model. Also, a higher degree of software testing is required in order to gather

sufficient data to apply to the model. For a more complete description of these models refer to [15, 13].

Another software reliability model is presented in Rome Laboratory's *Reliability Engineer's Toolkit* [25:124]. This particular model first calculates an initial software failure rate based on an initial estimate of the number of faults within the program. By assuming that a certain amount of time is allotted to testing and debugging the program, a growth model is then used to calculate an improved failure rate. This model is very general in nature; however it can serve as a starting basis for a reasonable estimate of software reliability.

2.6 Human Reliability

In many previous system reliability studies, the human component of the system is overlooked as having any significant contribution towards system reliability. However, almost all systems are interconnected with human links and rely in some way on human performance. Thus, the reliability evaluation of most engineering systems would be incomplete if human errors were not considered. Dhillon estimates that 20-30% of system failures are directly or indirectly related to human error [7:77]. Because of this more and more emphasis is being placed on modeling human reliability within systems.

Human reliability has been defined as the probability of accomplishing a job or task successfully by humans at any required stage in system operation within a specified time limit [7:3]. The ability to predict or model human reliability is dependent on a number of factors and is often a difficult task. The causes of human error may range from a number of factors, such as fatigue, stress, inadequate lighting, improper training,

distractions in work area, etc. These causes can be even further dissected to the level of the human body itself. Bodily systems such as the cardiovascular, muscular, digestive, or nervous system can also have an effect on the performance of the individual. The consequences of human error are dependent upon the particular task at hand, i.e. equipment being used, material being handled, etc. Due to the many variables involved in human reliability, it is apparent that the human reliability field is one in which the amount of emphasis placed on human performance will depend on the particular system to be analyzed.

2.6.1 Human Error Data Banks

In an effort to contribute toward the human reliability field, a number of data banks consisting of human reliability or failure rates have emerged. Many different data collection methods are used to construct these data banks. Dhillon groups these methods into four categories [7:169]. The first category is described as a direct manual method. This includes continuous observations, demonstrations, and sampled observations. The second category involves indirect manual methods such as proficiency tests and individual ratings. The third category includes system measurement records, i.e. maintenance reports, test records, etc. Finally the fourth category includes all automatic methods. Examples of automatic methods are physiological response recording instrumentation and task performance recording. Both the method used to collect the data and the type of data collected are important for determining which data banks may be appropriate for any specific system. While these data banks may be beneficial to the

human reliability field in general, finding a particular set of data to apply to a specific engineering system may be a difficult task.

2.6.2 Human Reliability Models

A general method often used to model human reliability involves selecting an appropriate probability distribution to represent the human's behavior and applying it to the basic definition of reliability [7:33]. Consider the basic definition of reliability in the form

$$R_e(t) = \exp\left[-\int_0^t h_e(t)dt\right] \quad (2.9)$$

where: $R_e(t)$ represents the human reliability at time t , and
 $h_e(t)$ represents the time-dependent human error (hazard) rate.

By collecting data on human performance, a particular statistical distribution can be selected to describe the human error rate. Some common distributions often used in reliability include the Weibull, normal, and exponential distributions. In many cases the exponential distribution is selected due to its ease of application. The exponential distribution implies that a constant human error (hazard) rate is present. If the exponential distribution is selected and the hazard rate is a constant, λ , then Equation 2.9 becomes

$$R_e(t) = \exp[-\lambda t] \quad (2.10)$$

Similar expressions can be derived using other probability distributions.

2.6.3 Human Error Estimation Techniques

Many human reliability analysis (HRA) models have been developed to aid in the quantification of human error. In general HRA attempts to predict human performance for a given situation. The HRA techniques presented here include: (1) the Ranked Data method, (2) Techniques for Human Error Rate Prediction (THERP), (3) the Success Likelihood Method (SLIM), and (4) the Human Error Assessment and Reduction Technique (HEART). These methods result in estimates of human error probabilities (HEP) or failure rates that can be incorporated into a system model. These methods may differ in applicability depending on what task or function the human is performing. The basic techniques for these methods are outlined below.

1. The Ranked Data technique has been carefully outlined by Seaver and Stillwell and can be found in [30]. The basic methodology of this technique is to divide the overall task into n individual events and assess a likelihood of error to each event. A brief description of this method is as follows:

- Divide the overall task into n individual events and use expert judgment to rank the events from least likely (rank=1) to most likely (rank= n) of being completed incorrectly.
- Create cumulative frequency table which shows the number of times each event was assigned up to a particular ranking.
- Divide table by total number of events to create table of proportions.
- Assuming that entries in the table of proportions are cumulative areas under the standard normal distribution, create table of normal deviates.
- Calculate row, column, and overall means for the table.
- Create subjective scale values by subtracting each row mean from the overall mean.

- Obtain estimates for two human error probabilities (HEP).
- Transform scale values into HEPs.

Moray [19] noted that Seaver and Stillwell's technique did not always restrict the resultant probabilities to the interval (0,1). He proposed an alternate approach for converting the subjective values to objective values of HEP. While this ranked data technique does provide valuable HEPs, the method requires probability estimates for two of the individual events within the overall task which may not be readily apparent. Also, the expert opinion used to determine the original ranking of these individual events introduces a certain degree of subjective error.

2. The Technique for Human Error Rate Prediction (THERP) method is regarded as the oldest and most widely accepted HRA technique [34]. In this method, the task under consideration is divided into individual events or actions and formulated as a probability tree model. Each action is categorized as belonging to one of two types of error. These error types are omission errors and commission errors. Omission errors occur when a procedure is not carried out and commission errors occur when a procedure is performed improperly. A database of HEPs is used to assign probabilities of error to each particular action. These HEPs can then be modified to account for specific performance shaping factors (PSF) related to the task. A PSF can be either internal or external factors which may effect an individual's performance. Examples of these factors include motivation, stress, work environment, and job skills. The THERP technique is primarily useful for human reliability prediction of discrete tasks [14].

3. The Success Likelihood Index Method (SLIM) places a considerable amount of weight on the PSFs which play a role in human error. This method attempts to identify the key PSFs involved in the performance of each individual event within the overall task [9:100-105]. These factors or influences are then ranked as multiples of the least important influence. The rankings of the influences are summed and normalized, i.e. each rank is divided by the sum. Each influence is assessed a particular quality or position in a spectrum of worst to best. Using the ranking and quality assignments, indexes are then calculated which can be used to determine HEP.
4. The Human Error Assessment and Reduction Technique (HEART) is similar to THERP in that HEPs are derived using generic data sources. It is similar to SLIM in that human reliability is assumed to be directly related to PSFs. Generic tasks and associated nominal HEPs are modified through the use of judgmental error producing conditions and associated multipliers. [17]

The HRA methods presented here are all directed towards quantifying human error. Although these methods are not an all inclusive list of human reliability estimation techniques, they do provide a reasonable representation of the most common methods used. Dhillon has presented a survey of human reliability literature which offers a more comprehensive listing of human reliability models [6]. HRA methods are generally quite subjective in nature. Expert judgments are relied upon at some point in almost every method to determine weights, rankings, estimated probabilities, etc. These judgments may create a certain amount of personal bias that is inserted into the HRA process.

Although HRA techniques may inherently have a certain amount of subjectivity associated with them, they do provide valuable insight to engineering systems dependent upon human performance.

2.6.4 Human Error Recovery

Human operators have the ability to prevent small errors from developing into a state of total system failure. These small errors may be due to the operator, as well as hardware or software components in the system. Error recovery may be possible regardless of the source of error. According to [28] there are three main phases of human error recovery. These phases include:

- (1) detection,
- (2) localization of causes, and
- (3) correction to normal.

In the first phase, the time between error occurrence and its detection is critical to provide even the possibility of recovery. In the second phase, the ability to locate the source of error will depend on the operator's reaction upon detection of an error. Finally, completion of the third phase is dependent on the probability of correction given that the error has been detected and localized. For many systems, errors are often unpredictable and may be unavoidable. The optimal design for a system may rely on the ability of a human operator to recover from small errors to some extent. If this is the case then an optimal interface should be developed between the human operator and the system in order to support the three phases of human error recovery.

2.7 Combinations of Reliability Models

Many techniques exist which enable the various component reliabilities to be incorporated into an overall system reliability model. The compatibility of component reliability models is important in the selection of an overall model. The techniques used to assess system reliability include Fault Tree Analysis, Markov models, and Monte carlo simulation. Fault Tree Analysis (FTA) is a method which utilizes a symbolic representation of the conditions that may cause a system to fail [23:172]. The first step in FTA is to identify particular failures of interest. The combinations of events that could lead to this failure are then explored. Refer to [23:172-178] for a further description of FTA.

Markov analysis is another technique used to evaluate system reliability. With a Markov modeling approach, all possible operating states of the system must be clearly defined. Each state is then categorized as being either a system working or a system failed state. Transition rates between states are used to develop a system reliability model. A continuous Markov model assumes constant transition rates. A semi-Markov analysis can be used to incorporate non-constant transition rates into the model. The main drawback to Markov type models is that the number of states for a system may lead to a high degree of computational complexity [21:128-134].

Monte carlo simulation is another means of assessing system reliability. In many cases analytical solutions may not be reasonable due to the size and complexity of the model. Monte carlo simulation involves a repeated evaluation of a logical model of the system where each run of the simulation uses randomly selected values for distributed

parameters [21:134]. System reliability can then be determined by taking the long term averages of the simulation output. For large models, Monte carlo simulation may involve extensive use of computer time.

2.7.1 Human-Machine Models

Recently there has been significant work done to model reliability within a human-machine system. A realistic system reliability model should include both hardware and human failure aspects. By including the human in the reliability model of the system many additional insights can be gained. Often, the human component of the system is just as important as the hardware or machine component.

A particularly interesting methodology is presented by Cacciabue, Carpignano, and Vivalda in [4]. This paper presents a technique for the analysis of human error, called DREAMS (Dynamic Reliability technique for Error Assessment in Man-machine Systems). This modeling technique attempts to account for human error tendencies as well as the ability to recover from potential errors. Within this dynamic approach, a continuous model is developed for system reliability.

A stochastic effectiveness model for human-machine systems was presented in [1]. This stochastic model accounts for a multiple-component human-machine system which may be required to perform several randomly arriving tasks. The components of the system can be represented as either hardware or software. This model was extended in [16] to analyze the transient behavior within a multiple-component human-machine system. This technique utilized computer simulation as a means of performing the analysis.

A general approach for reliability evaluation of redundant systems with human error is given by Dhillon in [7:77-138]. A Markov modeling approach is taken to analyze the reliability for specific cases of human-machine systems. A number of specific cases are considered, with Markov examples applied to each of them.

A number of models for system control have been developed to model a human controller. In [10] a stochastic sequential model for manually controlled man-machine tracking systems is presented. This model has the ability to account for delays and/or anticipations on the part of the controller. Another general model for human control of a dynamic system has been developed and is presented in [35]. This model attempts to describe decision making, planning, and continuous regulation within the control process.

This section has given a general representation of the work done in the human-machine reliability field. Many other important models have been developed for man-machine systems which include human reliability aspects. A literature survey by Lee, Tillman, and Higgins [14] includes many other references pertaining to the analysis of man-machine systems.

2.7.2 Robotics

There has been a fairly high amount of work concentrated on analyzing reliability within a robotic system. A book by Dhillon [8], *Robot Reliability and Safety*, provides a discussion of many topics related to robotic reliability. These topics include robot accidents, failure data, testing, robot safety, and human factors in robotics. Numerous examples of reliability models are also given. A Markov model is presented which represents a robot system that can fail due to human error as well as other errors [8:112].

Three system states are used for the model: normal operation, system failed due to human error, and system failed due to other than human error. The model assumes constant failure and repair rates between states. Expressions for system reliability are then developed.

A study by Schneider takes a more complete approach to assessing robotic system reliability [29]. The robotic system is divided into hardware, software, and kinematic components. Discussions regarding each of these components are presented. Throughout the study, a modular approach to robot design is emphasized. The techniques presented by [3] are utilized to determine kinematic reliability. The hardware and software components are represented together with a constant failure rate. Using a Monte carlo simulation, system reliability is then evaluated.

2.8 Summary

This literature review has presented many articles and books related to the four reliability components to be incorporated into a system reliability model for a teleoperated robotic manipulator. Many techniques and methods have been shown which are directed towards determining system reliability.

The next phase of this project will involve selecting appropriate reliability models for each of the four components. An appropriate method to combine all four reliability components into a system level model will also be determined. Once this system level model has been developed, further analysis will be aimed towards the NGMH/ATD system.

Chapter 3: Model Formulation

3.1 Introduction

This chapter describes the reliability model formulation for a general telerobotic system. The main components of a telerobotic system are analyzed and appropriate models for each are selected. Since these components interact to form the overall telerobotic system, the component models are then combined into system level reliability models. The key to developing a good system reliability model is to select component reliability models which can be integrated into an overall system model.

There are two main models investigated throughout this chapter. The first model is a simple first-order representation of the system. This model serves as a starting basis for further study of the system reliability. A Markov modeling approach was taken to represent this first-order model. Using this approach, both analytical and simulation results were able to be investigated. The second model is an expanded version of the first-order model. In the expanded model, the overall task of the system is divided into a number of subtasks. A first-order Markov model is then applied to each of the subtasks. The overall system reliability is then obtained by combining the results from each Markov model. This expanded model can be used to include a higher detail level within the system. Using simulation, a semi-Markov model can be developed which has component failure distributions other than the exponential. Numerical examples of both models are given and important aspects of each model will be discussed.

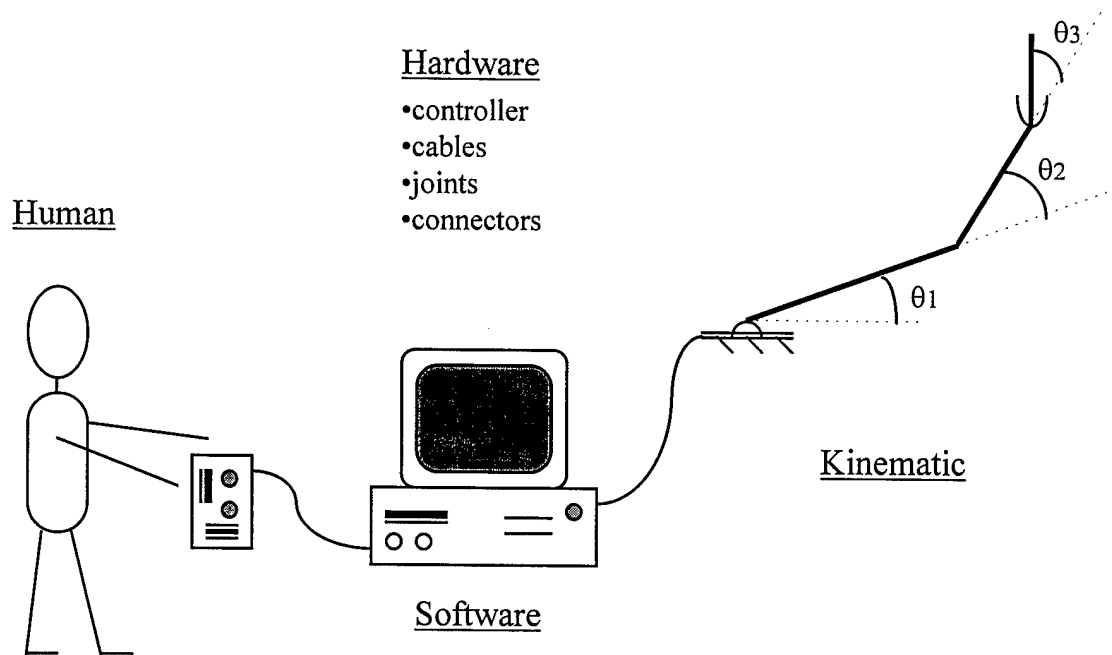


Figure 3.1 Schematic of Telerobotic System

3.1.1 General Telerobotic System Description

In general, a teleoperated robotic system consists of four main components which affect system reliability. These components are: (1) human, (2) software, (3) hardware, and (4) kinematic. Refer to Figure 3.1 to see a schematic of a telerobotic system that depicts these four main components. Notice that these components must all interact within the telerobotic system. While these components should each be designed to have a high degree of reliability, they should also be well integrated within the system so as to maximize overall system reliability. Each component has inherent reliability characteristics which must be carefully explored in order to develop a representative system reliability model. A brief review of each of these components is given below.

Human Reliability

Humans typically introduce a great deal of uncertainty in system reliability models. Because the causes of human errors (fatigue, stress, improper training, etc.) are so difficult to quantify, and since no two individuals are exactly alike, human reliability estimation is generally quite subjective in nature. Experimentation is often required in order to develop a good human reliability model for a specific system.

Software Reliability

Software reliability can be highly variable from one system to the next. The amount of software and the complexity of the software used by a system varies greatly from one system to the next. For a telerobotic system this is directly related to the degree of autonomy given to the system. Unlike most reliability components, software does not degrade with time. Errors in software programming are there from the start. However, it is often very difficult to predict errors within a particular program. Extensive testing is often required to produce a good software reliability model.

Hardware Reliability

Hardware is generally the most common subject of a system reliability analysis. A great deal of data has been collected on most types of hardware, both electrical and mechanical. Rome Laboratory's *Non-Electronic Parts Reliability Data* and MIL-HDBK-217F are two sources of hardware data [24,18]. Unique hardware components are generally tested to obtain an estimate of reliability if there is no data available.

Kinematic Reliability

Kinematic reliability is the probability of the end-effector being within a certain error bound of its desired position and orientation (pose) [3]. This component is directly dependent on the kinematic configuration of the manipulator as well as variations in joint accuracy. The tolerances used to define the error bound also play a large role in the kinematic reliability determination.

3.2 First-Order Markov Model

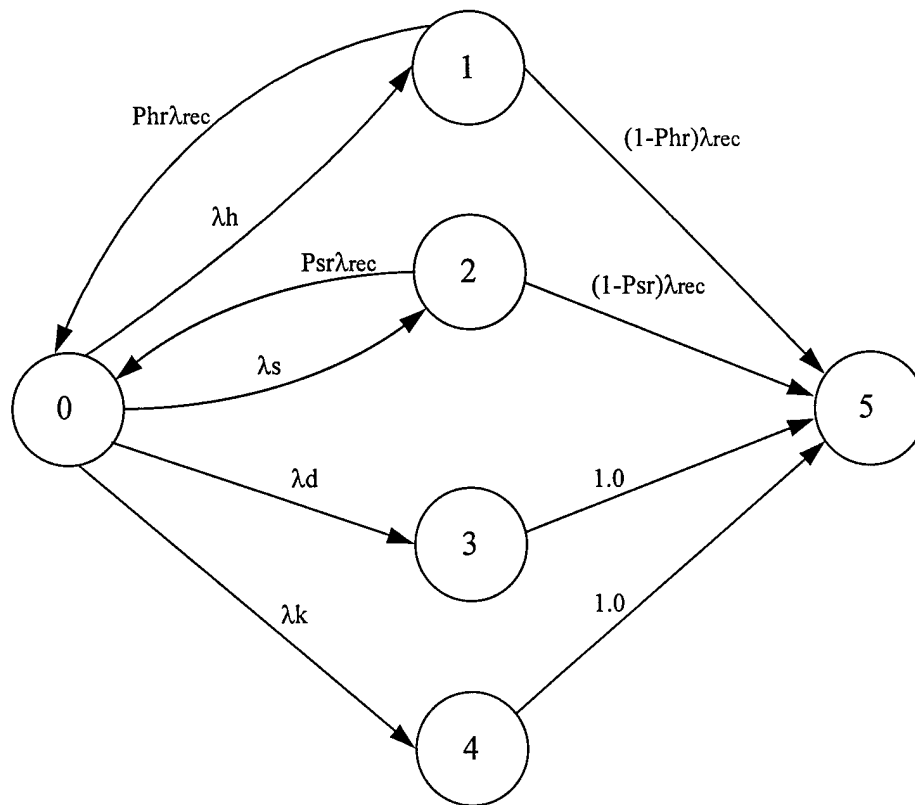
A first-order model of a general teleoperated robotic system was developed to provide a starting basis for a more complex model to be developed later. For this first-order model, each of the four main components is assumed to have a constant failure rate. Thus the reliability for any component can be expressed as

$$R_i(t) = \exp(-\lambda_i t) \quad (3.1)$$

where λ_i represents the constant failure rate for the i th component. Also the Mean Time To Failure (MTTF) for each component is easily obtained through the relationship

$$MTTF = \frac{1}{\lambda_i} \quad (3.2)$$

Assuming that all components possess a constant failure rate may or may not be a good assumption. However, this assumption is generally acceptable within a first-order model. Having constant failure rates also allows the use of a simple Markov model to combine all components into an overall system model.



- State 0 = normal operation
- State 1 = human error
- State 2 = software error
- State 3 = hardware error
- State 4 = kinematic error
- State 5 = system failure

Figure 3.2 First-Order Markov Model

The first-order telerobotic system reliability model is shown in Figure 3.2. This Markov model requires constant failure rates for each of the four components. These failure rates each lead to an error state for one of the components. The telerobotic system may recover from both human and software errors while a hardware or kinematic error is assumed to lead directly to system failure. The recovery from human or software errors is

represented as a probability of recovery multiplied by a recovery rate. The probability of recovery can be easily altered within the model to explore the effects of different recovery probabilities. Also note that the probability of recovery for human and software errors does not need to be the same. The recovery rate used should represent the constant rate at which recovery occurs. It can be chosen large enough to represent instantaneous recovery if desired. In the model developed here, the recovery rates for both a human and software error are assumed to be the same. A summary of the variables included in the first-order model is shown in Table 3.1. Once these variables have been specified, this first-order Markov model can be solved analytically or using a simulation.

Table 3.1 Variables of First-Order Markov Model

Variable	Description
λ_h	Human failure rate
λ_s	Software failure rate
λ_d	Hardware failure rate
λ_k	Kinematic failure rate
λ_{rec}	Recovery rate
P_{hr}	Probability of recovery given human error
P_{sr}	Probability of recovery given software error

3.2.1 Example of First-Order Markov Model

In order to illustrate the use of the first-order Markov model a numerical example is given below. Since each component of the system is assumed to have a constant failure rate, these rates play a large role in the outcome of the model. Each component's failure distribution (exponential) is completely specified by its failure rate. In this particular example, arbitrary failure and recovery rates were selected based on reasonable values

and relative significance. For example, intuitively, the human failure rate should be higher than the hardware failure rate. A reasonable hardware failure rate was chosen to be 500 failures per 10^6 operating hours. The human failure rate was then estimated to be ten times the hardware failure rate. The recovery rate was chosen to be large enough to correspond to instantaneous recovery from both human or software errors. The probability of recovery from either a human or software error was chosen to be 0.5. Refer to Table 3.2 for a complete list of values used within this example. Please note that the numerical values chosen to represent each of the variables are not meant to be a predictive measure of any particular system, but rather simply to illustrate the use of the first-order Markov model. The time at which reliability is estimated was chosen arbitrarily to be 15 hours. Using this first-order Markov model (Figure 3.2) the system reliability was estimated at 15 hours both analytically and through the use of a computer simulation.

Table 3.2 Values used in First-Order Markov Model Example

Variable	Value (λ s in number/hour)	Description
λ_h	0.005	Human failure rate
λ_s	6.667E-4	Software failure rate
λ_d	5.0E-4	Hardware failure rate
λ_k	0.001	Kinematic failure rate
λ_{rec}	36.0	Recovery rate
P_{hr}	0.5	Probability of recovery given human error
P_{sr}	0.5	Probability of recovery given software error

3.3.2 Results from First-Order Markov Model Example

Analytic Solution

The analytic solution for the first-order Markov model involves solving a system of differential equations for the Markov process. The differential equations describing the various states within the model can be written as

$$\begin{aligned}\frac{dP_0(t)}{dt} &= 0.5\lambda_r P_1(t) + 0.5\lambda_r P_2(t) - (\lambda_h + \lambda_s + \lambda_d + \lambda_k)P_0(t) \\ \frac{dP_1(t)}{dt} &= \lambda_h P_0(t) - (0.5\lambda_r + 0.5\lambda_r)P_1(t) \\ \frac{dP_2(t)}{dt} &= \lambda_s P_0(t) - (0.5\lambda_r + 0.5\lambda_r)P_2(t) \\ \frac{dP_3(t)}{dt} &= \lambda_d P_0(t) - \lambda_r P_3(t) \\ \frac{dP_4(t)}{dt} &= \lambda_k P_0(t) - \lambda_r P_4(t) \\ \frac{dP_5(t)}{dt} &= 0.5\lambda_r P_1(t) + 0.5\lambda_r P_2(t) + \lambda_r P_3(t) + \lambda_r P_4(t)\end{aligned}\tag{3.3}$$

where $P_i(t)$ represents the probability of being in state i at time t .

Assuming that at $t=0$ the system is in the normal operating state (i.e. $P_0(0)=1$), the Laplace Transform of each of these differential equations results in the following system of equations:

$$\begin{aligned}sf_0(s) - 1 &= 0.5\lambda_r f_1(s) + 0.5\lambda_r f_2(s) - (\lambda_h + \lambda_s + \lambda_d + \lambda_k)f_0(s) \\ sf_1(s) &= \lambda_h f_0(s) - \lambda_r f_1(s) \\ sf_2(s) &= \lambda_s f_0(s) - \lambda_r f_2(s) \\ sf_3(s) &= \lambda_d f_0(s) - \lambda_r f_3(s) \\ sf_4(s) &= \lambda_k f_0(s) - \lambda_r f_4(s) \\ sf_5(s) &= 0.5\lambda_r f_1(s) + 0.5\lambda_r f_2(s) + \lambda_r f_3(s) + \lambda_r f_4(s)\end{aligned}\tag{3.4}$$

Using this system of equations, $f_0(s)$ can be expressed as

$$f_0(s) = \frac{s + \lambda_r}{(s + \lambda_r)(s + \lambda_h + \lambda_s + \lambda_d + \lambda_k) - 0.5\lambda_r(\lambda_h + \lambda_s)} \quad (3.5)$$

The inverse Laplace transform of this expression provides the probability of being in state 0 at time t , $P_0(t)$. Since state 0 represents normal operation of the system, this is also the reliability of the system at time t .

$$R(t) = P_0(t) \quad (3.6)$$

Through the use of MathCAD the solution was found to be $R(15)=0.937$. The detailed calculations from this example can be found in Appendix A.

A brief sensitivity analysis was conducted on the first-order model by varying the probability of recovery for human and software errors. Figure 3.3 depicts the results obtained when these probabilities were varied from 0.0 to 1.0. The heavy solid line represents the case when P_{sr} and P_{hr} were set equal and varied from 0.0 to 1.0. The thin gray line represents the case when P_{sr} was held constant at 0.5 and P_{hr} was varied from 0.0 to 0.1. From this plot it is clearly seen that the probability of recovering from a human error has the largest impact on system reliability. The case where both P_{sr} and P_{hr} are varied is almost identical to holding P_{sr} constant and only changing P_{hr} . This results directly from the failure rates used within the model. Since the human failure rate is much higher than the software failure rate, the probability of recovering from a human error should have a larger impact on the system reliability.

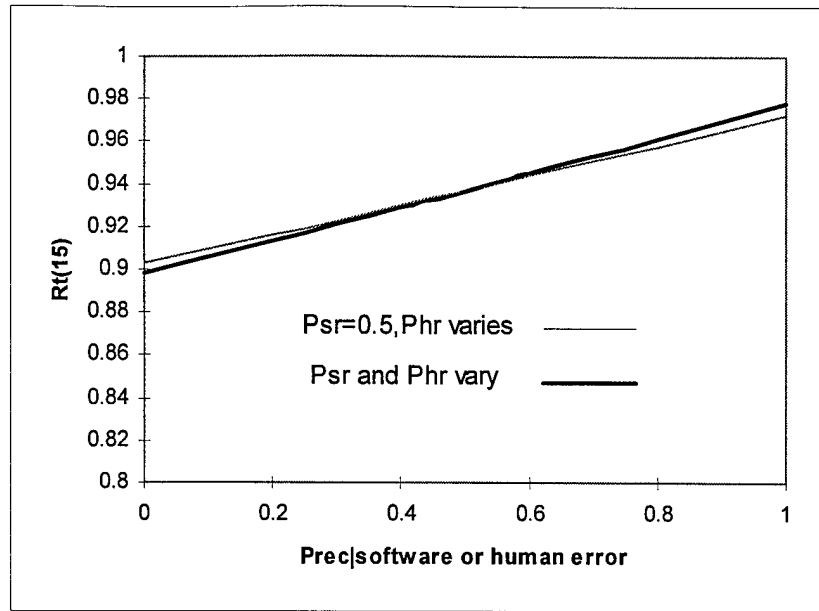


Figure 3.3 Sensitivity Analysis on Probability of Recovery

Simulation

A SLAM II simulation program was utilized to conduct a simulation of the first-order Markov model. Each simulation run stopped when either the 15 hour time limit was reached or a system failure occurred. The reliability of the system at 15 hours was then calculated as

$$R(15) = \frac{\text{Total number of runs} - \text{Number of failures}}{\text{Total number of runs}} \quad (3.7)$$

The total number of simulation runs was chosen to be 5000. Theoretically, the steady state solution of the simulation should converge with the analytical solution. After 5000 runs the solution was calculated to be $R(15)=0.9158$. A more detailed presentation of this simulation including SLAM II output can be found in Appendix A.

The reliability value calculated using the simulation method was slightly less than the analytical solution. This is most likely attributed to the number of runs chosen for the simulation. A higher number of runs should result in a value closer to the value of the analytic solution. An infinite number of runs should agree exactly with the analytic solution. Due to time constraints, the simulation results were deemed to be sufficiently close to the analytical solution for the purposes of this example.

3.3 Expanded Markov Reliability Model

The next phase of the telerobotic reliability development involves expanding the first-order model into a more complete system reliability model. The first step in creating this expanded model entails dividing the overall task of the system into n distinct subtasks. Each subtask should include all four of the system components. Within each subtask, failure rates are then estimated for each of the four components. The recovery probabilities within each subtask should also be estimated. The rates and probabilities for each subtask may or may not be the same. Once failure rates and recovery probabilities have been specified for each subtask, the first-order Markov model presented earlier can be used to find reliability estimates for each of the subtasks. Using the first-order model, the reliability for each subtask can be solved analytically or through a computer simulation.

Each subtask must be completed satisfactorily in order for the overall task to be a success. Therefore each subtask can be thought of as contributing to a series system

reliability model. The overall system reliability can then be calculated using the expression

$$\begin{aligned} R_{sys}(T) &= R_1(T_1) \cdot R_2(T_2) \cdot R_3(T_3) \cdots R_n(T_n) \\ T &= T_1 + T_2 + T_3 + \cdots + T_n \end{aligned} \quad (3.8)$$

where $T_i, i=1, \dots, n$, represents the total amount of time spent within each subtask i . The following numerical example attempts to illustrate the use of this expanded reliability model.

3.3.1 Example of Expanded Markov Reliability Model

For the numerical example presented here, a three link manipulator will be used for the calculations (see Figure 3.4). The three link manipulator is assumed to perform some type of pick and place operation. This is a common task for robotic manipulators used in assembly operations [12:67]. Due to the repetitive nature of assembly operations, robots are generally well suited for these tasks. If a generic pick and place operation is to be performed by the three link manipulator, the overall task of picking up and object and placing it can be divided into the following subtasks:

1. Pick up object
2. Positioning of manipulator
3. Orientation of manipulator
4. Place object
5. Return to starting location

Each of these subtasks should be analyzed to determine their dependence on each of the four main components. Also each component may have subcomponents that should be investigated. For example, consider a hardware system. For a telerobotic

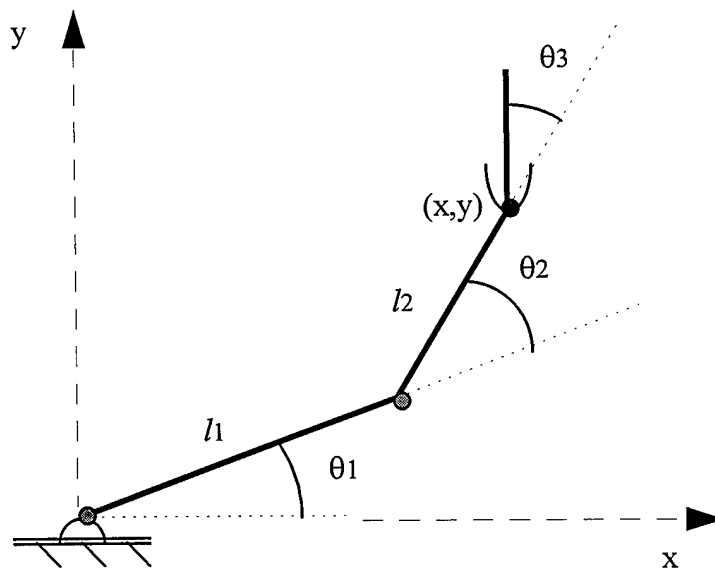


Figure 3.4 Three link manipulator used for example calculations

manipulator, this hardware system can be quite complex, consisting of both electrical and mechanical components. The components may or may not be required for each of the subtasks to be completed successfully. Some components may be used throughout the overall task of the manipulator while others may only be used for particular subtasks within the overall task. Careful judgment must be employed while choosing which components are required for completion of each subtask. Each of the four main components will now be investigated to determine failure rates for the expanded model.

Human

In order to assign failure rates to the human component of the system for each subtask, subjective error probabilities should be carefully assessed. The human error probabilities (HEPs) used in this example were estimated based on relative difficulty

between subtasks. The HEPs shown here are subjective estimates, only meant to illustrate the use of the model. In order to develop failure rates to be inserted into the model, each HEP was transformed into a failure rate using the approximation

$$\lambda_h \approx \frac{HEP}{t_i} \quad (3.9)$$

where t_i is the time to perform subtask i . The Table 3.3 provides the results of this assessment. Ideally, a testing procedure could be utilized to develop accurate HEPs for specific subtasks within a particular system. Also, human reliability databases could be employed to assign HEPs to subtasks within a system.

Table 3.3 Human error probabilities (HEP) for each of the subtasks

Subtask	Time for subtask (seconds)	HEP	λ_h (failures per sec)
Pick up object	5	0.0001	2.0E-5
Positioning of manipulator	20	0.005	2.5E-4
Orientation of manipulator	15	0.002	1.333E-4
Place object	5	0.0005	1.0E-4
Return to starting location	15	0.0002	1.333E-5

Software

The software failure rate was calculated using the Littlewood Modular Software Reliability model [15]. Generally, this model would make use of data obtained during module development and testing. For this particular example, the following information is assumed to be known concerning the software program. A modular software structure is evident which consists of six modules as shown in Table 3.4. Each time the program is used, the computation module for each joint is called 50 times, the input/output (I/O)

module is called twice and the memory manager module is called twice. Also, each time a joint computation module is called, it calls the I/O and the memory manager once. Control is always returned to the module from which control originated. Using this information, there will be 908 transitions between modules each time the program is used. Refer to Table 3.5 for a listing of control transfer frequencies.

Table 3.4 Example Software Module Characteristics

Module	Function	a_i	v_i (failures/ 10^6 hours)
1	Supervisor	0.8333	50
2	Joint 1 Computation	0.20	200
3	Joint 2 Computation	0.20	200
4	Joint 3 Computation	0.20	200
5	Input/Output	0.2333	100
6	Memory Manager	0.0833	50

Table 3.5 Example Software Control Transfer Frequencies

$i-j$	b_{ij}	$i-j$	b_{ij}	$i-j$	b_{ij}	$i-j$	b_{ij}	$i-j$	b_{ij}	$i-j$	b_{ij}
1-1	0.0	2-1	0.0551	3-1	0.0551	4-1	0.0551	5-1	0.0022	6-1	0.0022
1-2	0.0551	2-2	0.0	3-2	0.0	4-2	0.0	5-2	0.0551	6-2	0.0551
1-3	0.0551	2-3	0.0	3-3	0.0	4-3	0.0	5-3	0.0551	6-3	0.0551
1-4	0.0551	2-4	0.0	3-4	0.0	4-4	0.0	5-4	0.0551	6-4	0.0551
1-5	0.0022	2-5	0.0551	3-5	0.0551	4-5	0.0551	5-5	0.0	6-5	0.0
1-6	0.0022	2-6	0.0551	3-6	0.0551	4-6	0.0551	5-6	0.0	6-6	0.0

Applying the data shown in Tables 3.4 and 3.5 into Equation 2.8 the following calculation can be made

$$\lambda_s = \sum_i a_i v_i + \sum_i b_{ij} \lambda_{ij}$$

$$\lambda_s = (0.0833)(50) + (0.20)(200) + (0.20)(200) + (0.20)(200) + (0.2333)(100) + (0.0833)(50) + 18[(0.0551)(1.0)] + 4[(0.0022)(1.0)]$$
(3.10)

$$\lambda_s = 152.667 \text{ failures}/10^6 \text{ hours} = 4.24 \text{ E-8 failures/second}$$

In the above calculation, all λ_{ij} s were assumed to be 1.0 failure/second. Please note that the failure rates used within the Littlewood model are simply being used to illustrate the use of the expanded model. The software failure rate was assumed to be constant throughout the overall task. Therefore each subtask has the same failure rate.

Hardware

The hardware failure rate was obtained by selecting the major hardware components from the telerobotic system. For this example, these components were chosen to be the three joint motors and a joystick and switch used to control the device. Thus, using a simple series representation, the hardware failure rate can be calculated using the following relationship

$$\lambda_d = \lambda_{j1} + \lambda_{j2} + \lambda_{j3} + \lambda_{switch} + \lambda_{joystick} \quad (3.11)$$

These rates were estimated using the NPRD handbook [24]. Table 3.6 summarizes the hardware failure rate values used in this example.

Using the failure rates shown in Table 3.6, the combined hardware failure rate was then calculated to be

$$\begin{aligned} \lambda_d &= [3(14.4367)+4.8+3.2285] = 51.3386 \text{ failures}/10^6 \text{ hours} \\ &= 1.43 \text{ E-8 failures/second} \end{aligned} \quad (3.12)$$

This hardware failure rate was assumed to be constant throughout the overall task of the manipulator. Therefore, each subtask was assumed to have the same hardware failure rate.

Table 3.6 Hardware failure rate values

Hardware Component	Failure rate, λ (failures per 10^6 hours)
Electrical Motor, DC \rightarrow j1	14.4367
Electrical Motor, DC \rightarrow j2	14.4367
Electrical Motor, DC \rightarrow j3	14.4367
Switch	4.8
Joystick	3.2285

Kinematic

The kinematic reliability for each subtask was obtained through the use of the method introduced by Bhatti and Rao [3]. The position of the manipulator is determined by the (x,y) coordinates at the third joint (refer to Figure 3.4). The equations for the position are

$$\begin{aligned} x &= l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ y &= l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \end{aligned} \quad (3.13)$$

where l_1 and l_2 represent the lengths of the first and second links respectively, and θ_1 and θ_2 are the first and second joint angles. The orientation of the manipulator is based on the angle of the end effector and can be calculated as

$$\Phi = \theta_1 + \theta_2 + \theta_3 \quad (3.14)$$

Each joint angle, θ , is assumed to be a normal random variable with a given mean and standard deviation. The mean is specified as the desired angle for each joint, while the standard deviation, σ , is assumed to be 0.01° for each joint. Using a computer simulation, the mean value for each joint angle is specified and the position and orientation resulting from these angles are calculated. If the position and orientation

(pose) are outside of a specified error bound from the desired pose, then a kinematic failure has occurred for that particular simulation run. The tolerances used in this example are $\pm 1\text{mm}$ for the x and y coordinates and $\pm 0.05^\circ$ for the orientation angle. The kinematic reliability for a specified location along a given trajectory can then be calculated as

$$R_k = \frac{\text{Total number of runs} - \text{Number of failures}}{\text{Total number of runs}} \quad (3.15)$$

The kinematic reliability is summarized in the following probability statement

$$R_k(x, y, \Phi) = P \left\{ \begin{array}{l} (x_d - 0.001 \leq x \leq x_d + 0.001) \cup \\ (y_d - 0.001 \leq y \leq y_d + 0.001) \cup \\ (\Phi_d - 0.05 \leq \Phi \leq \Phi_d + 0.05) \end{array} \right\} \quad (3.16)$$

A FORTRAN program was developed based on the work previously done by Schneider [29] to calculate the R_k values. This simulation was performed 1000 times to determine a kinematic reliability estimate for each position along the straight line trajectory illustrated in Figure 3.5. The solid lines shown in Figure 3.5 represents the pose of the manipulator at the starting location. Dashed lines attempt to illustrate the pose of the manipulator as it moves along its trajectory. Through the use of Equations 3.12 and 3.13, the three desired joint angles are calculated for any specified pose. Table 3.7 contains a summary of information related to the position and orientation of the manipulator at each location along this trajectory.

For subtasks that involve more than one pose (i.e. positioning of manipulator), the kinematic reliability was taken to be the minimum R_k among the poses utilized. Once the kinematic reliability estimates have been obtained for each subtask, the kinematic failure

rate for each subtask can be approximated in a manner similar to that used to calculate the human failure rate (Equation 3.9). Since the R_k value obtained for each pose is a point estimate of kinematic reliability, it is proposed to calculate the kinematic failure rate as

$$\lambda_k \approx \frac{1 - R_k}{T_i} \quad (3.17)$$

where T_i represents the total time spent performing subtask i .

For the first subtask (pick up object) the manipulator is assumed to remain in its starting pose. The value of R_k at this location was found to be 0.993. Since this subtask is assumed to take 5 seconds, the kinematic failure rate for a single pick and place operation for subtask 1 is approximated by

$$\lambda_{k1} \approx \frac{1 - 0.993}{5} = 0.0014 \text{ failures/second} \quad (3.18)$$

If multiple pick and place operation are to be evaluated, then the kinematic failure rate can be estimated by dividing through with the total amount of time for each subtask. For example, if 50 pick and place operations are to be performed, then the total amount of time spent in subtask 1 is 250 seconds. The kinematic failure rate for subtask 1 can now be calculated as

$$\lambda_{k1} \approx \frac{1 - 0.993}{250} = 2.8E-5 \text{ failures/second} \quad (3.19)$$

The failure rates used in this example for both a single task and 50 repetitions are summarized in Table 3.8.

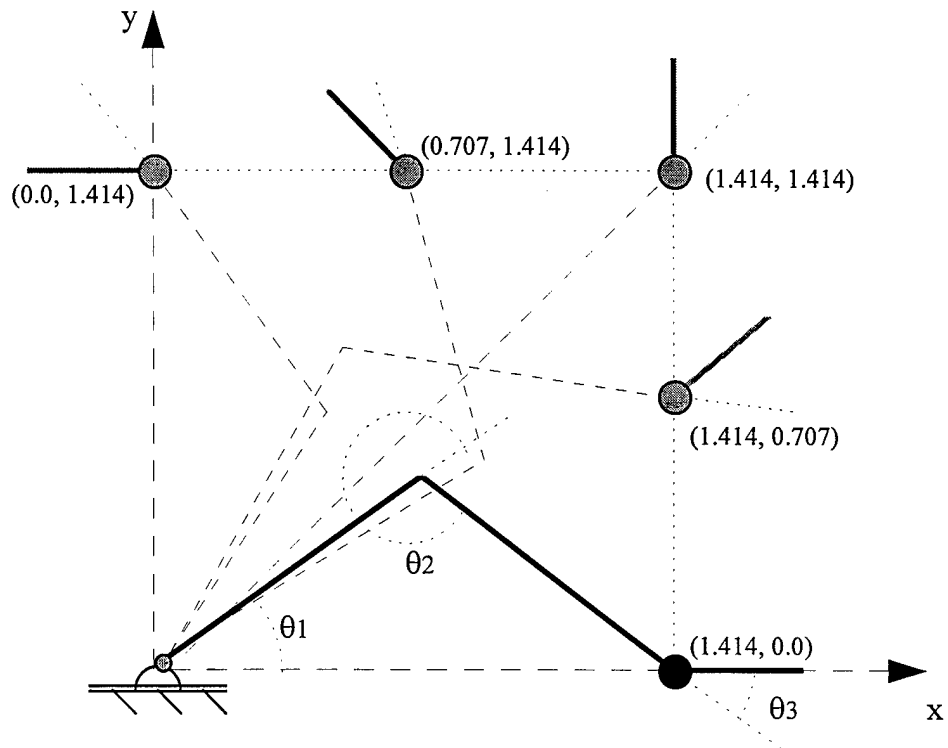


Figure 3.5 Straight line trajectory of three link manipulator

Table 3.7 Positions along straight line trajectory

location number	(x,y) location	θ_1 (degrees)	θ_2 (degrees)	θ_3 (degrees)
1	(1.414,0.0)	62	266	32
2	(1.414,0.707)	77.334	281.537	46.129
3	(1.414,1.414)	45	0	45
4	(0.707,1.414)	12.666	78.463	43.871
5	(0.0,1.414)	28.126	93.823	58.051

Table 3.8 Kinematic Failure Rate Estimates

Subtask	R_k	λ_k (single task) failures/second	λ_k (50 repetitions) failures/second
Pick up object	0.993	0.0014	2.8E-5
Positioning	0.990	5.0E-4	1.0E-5
Orientation	0.993	4.67E-4	9.333E-6
Place object	0.993	0.0014	2.8E-5
Return	0.990	6.667E-4	1.333E-5

3.3.2 Results from Expanded Markov Model Example

The expanded model was utilized to estimate the system reliability after 50 repetitions of a pick and place operation. Since each pick and place operation is assumed to take 60 seconds, 50 repetitions would require 3000 seconds or 50 minutes. Reliability estimates were obtained for each subtask both analytically and through a simulation. The probability of recovery from either a human error or software error was assumed to be 0.5 for all subtasks. Table 3.9 summarizes the failure rates, total time for each subtask, as well as the reliability estimates for each subtask.

Table 3.9 Summary Table for Reliability After 50 Repetitions (λ s in failures per second)

Subtask	λ_h	λ_s	λ_d	λ_k	Total time, sec	R_i <i>analytic</i>	R_i <i>simulation</i>
1	2.0E-5	4.24E-8	1.43E-8	2.8E-5	250	0.9896	0.990
2	2.5E-4	4.24E-8	1.43E-8	1.0E-5	1000	0.8641	0.825
3	1.333E-4	4.24E-8	1.43E-8	9.333E-6	750	0.9386	0.917
4	1.0E-4	4.24E-8	1.43E-8	2.8E-5	250	0.9762	0.975
5	1.333E-5	4.24E-8	1.43E-8	1.333E-5	750	0.9844	0.981
R_{TOTAL}						0.77132	0.71636

Notice that in general, the simulation results seem to correspond fairly well with the analytical solution for each subtask. The overall reliability estimate obtained using simulation is slightly lower than the analytical estimate. This can be attributed to the number of runs used to simulate each subtask. In this example 1000 simulation runs were made for each subtask. A higher number of runs should result in a solution closer to the

analytical result. For the purposes of this example, the simulation result seems to be close enough to assume agreement between the two solution methods.

The simulation results enable the modeler to determine which system components are causing the system failure. The output from the simulation for each subtask depicts the number of human recoveries, human failures, software recoveries, software failures, hardware failures, and kinematic failures. Table 3.10 provides a summary of the simulation results for each subtask.

Table 3.10 Summary of Simulation Results (1000 runs per subtask)

Subtask	Human Recovery	Human Failure	Software Recovery	Software Failure	Hardware Failure	Kinematic Failure	Total Failures
1	1	5	0	0	0	5	10
2	160	166	0	0	0	9	175
3	63	78	0	0	0	5	83
4	13	20	0	0	0	5	25
5	3	10	0	0	0	9	19

Notice that the reliability estimates are dominated by human and kinematic failures at each subtask. This is directly attributed to the failure rates estimated for these components. Since their failure rates are considerably higher than the software and hardware failure rates, the human and kinematic failures should dominate the model.

A brief sensitivity analysis was conducted on the probability of recovery from either a human or software error using the analytical model. This is the same analysis previously conducted on the first-order model. This sensitivity analysis is depicted in Figure 3.6. The solid line represents the case when both P_{hr} and P_{sr} are varied from 0.0

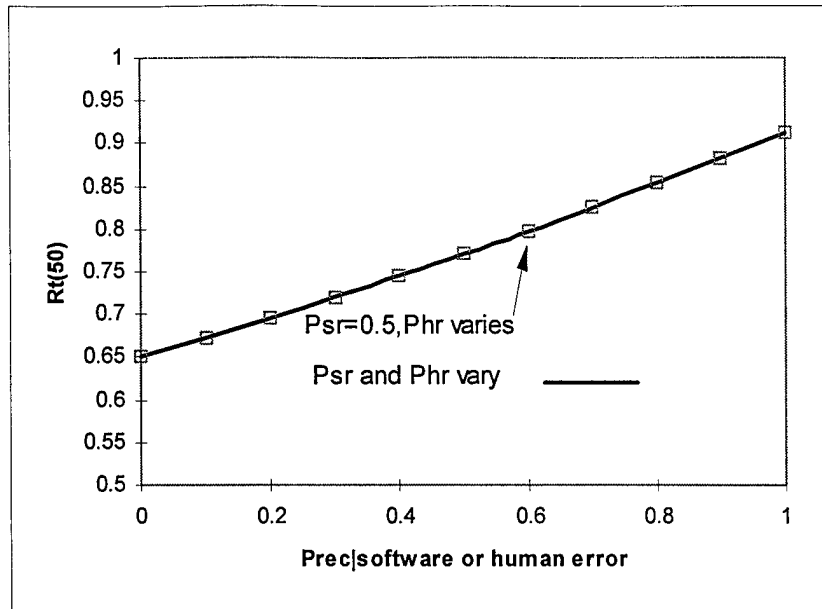


Figure 3.6 Sensitivity Analysis on Probability of Recovery

to 1.0. The points that appear to lie directly on this line represent the case when P_{sr} is held constant at 0.5 and P_{hr} is varied from 0.0 to 1.0. Notice that the human failure rate seems to be dictating the outcome of the model. This agrees with the results obtained from the simulation.

3.4 Expanded Semi-Markov Reliability Model

Since most hardware components exhibit some form of wear leading to failure, a constant failure rate model may not be sufficient to model the hardware component. In an attempt to account for some of the wear on the system which is typical of most hardware components, an increasing failure rate was introduced for the hardware components. The introduction of a non-constant failure rate results in a semi-Markov model for each subtask. The hardware failure distribution was assumed to follow a Weibull distribution.

With the exponential distribution, only the failure rate, λ , needs to be specified. For a Weibull distribution, both a shape parameter, β , and a scale parameter, α , must be specified. If the MTTF for both the exponential and Weibull distributions are assumed to be equal, then the parameters from these distributions can be related by

$$MTTF = \alpha \Gamma\left(1 + \frac{1}{\beta}\right) \Rightarrow \frac{\alpha}{MTTF} = \left[\Gamma\left(1 + \frac{1}{\beta}\right)\right]^{-1} \quad (3.20)$$

The value for the shape parameter, β , was chosen to be 2.0. The scale parameter, α , can now be calculated as

$$\alpha = 1.128 \left(\frac{1}{1.43E-8} \right) = 79,124,692 \text{ seconds} \quad (3.21)$$

The system reliability was then estimated after 50 repetitions, using a Weibull distribution to model the hardware failures. The simulation results from this semi-Markov model including a Weibull distribution for the hardware component are shown Table 3.11. Notice that these results appear identical to the results depicted in Table 3.10, with the exception of the hardware failures. Because each subtask is being modeled as having a Weibull distributed hardware failure distribution, the increasing failure rate is

Table 3.11 Summary of Simulation Results (Weibull hardware, 1000 runs per subtask)

Subtask	Human Recovery	Human Failure	Software Recovery	Software Failure	Hardware Failure	Kinematic Failure	Total Failures
1	1	5	0	0	1	5	11
2	160	166	0	0	12	9	187
3	63	78	0	0	7	5	90
4	13	20	0	0	1	5	26
5	3	10	0	0	8	9	27

causing hardware failures to appear. More hardware failures occur as expected which decreases the overall reliability estimate for the system. The reliability estimate after 50 repetitions has now decreased to $R(50)=0.6934$.

3.5 Reliability Index Development

Since the method used to calculate the kinematic reliability values provides point estimates of reliability for specified configurations, transforming these point estimates to failure rates (refer to Equation 3.17) may not provide the best representation of the kinematic component. In order to maintain these point estimate values, a reliability index is developed.

The reliability index is based on a combination of two models. A semi-Markov model is used for the human, software, and hardware components of a telerobotic system as before (with the kinematic component removed). A separate simulation model is used to calculate kinematic reliability values for various configurations of the manipulator. A reliability index can then be developed for each subtask as follows

$$R_{ji} = R(T_i) \cdot R_{ki} \quad (3.22)$$

where R_{ji} represents the reliability index for the i th subtask,
 $R(T_i)$ is the reliability estimate from the semi-Markov simulation,
 T_i is the total time for subtask i in the simulation, and
 R_k is the kinematic reliability estimate for the i th subtask.

The R_{ki} values used for each subtask are the minimum kinematic reliability values for the subtask. Choosing the minimum R_{ki} values corresponds to the study conducted by Schneider [28]. Using the minimum R_{ki} values leads to a somewhat conservative

estimate of the kinematic component. Once a reliability index has been obtained for each subtask, an overall reliability index, R_I , can then be calculated using

$$R_I = R_{I1} \cdot R_{I2} \cdots R_{In} \quad (3.23)$$

where n is the number of subtasks.

Using the same example as before (refer to section 3.4), a semi-Markov simulation was conducted using only the human, software, and hardware components. Each subtask was simulated using 1000 runs. The results from the semi-Markov simulation are shown in Table 3.12. Notice that the kinematic component has been removed. Also note that the simulation results are very similar to those shown in Table 3.11 for the other components.

Table 3.12 Summary of Semi-Markov Simulation Results

Subtask	Human Recovery	Human Failure	Software Recovery	Software Failure	Hardware Failure	Total Failures
1	1	4	0	0	1	5
2	160	166	0	0	12	178
3	64	80	0	0	7	87
4	13	20	0	0	1	21
5	3	10	0	0	8	18

The reliability index calculations for this example are shown in Table 3.13. The minimum R_k values used correspond to those found in Table 3.8. Recall that these values are based on point estimates obtained along the intended trajectory of the manipulator. The $R(T_i)$ values correspond to the results from the semi-Markov simulation for the human, software, and hardware components. From this brief analysis, it appears that

subtask 2, positioning of the manipulator, is the most critical subtask in this example.

Table 3.12 shows that this is largely due to the high number of human errors associated with this subtask.

Table 3.13 Reliability Index Calculations

Subtask, i	T_i	$R(T_i)$	Min R_{ki}	R_{Ii}
1	250	0.995	0.993	0.988
2	1000	0.822	0.990	0.814
3	750	0.913	0.993	0.907
4	250	0.979	0.993	0.972
5	750	0.982	0.990	0.972
				0.689

3.6 Summary

A general telerobotic reliability model has been developed through a series of iterative models. A first-order Markov model was introduced as a basis for further reliability models. An expanded Markov modeling technique was then presented which divided the overall task of the telerobotic system into a number of subtasks. Next, non-constant failure rates within the hardware component of the system were introduced to the expanded modeling approach by means of a semi-Markov model. Finally, a reliability index was developed which combines the results of a semi-Markov model for the human, software, and hardware components with point estimates of kinematic reliability values.

The general telerobotic reliability models developed in this chapter are not intended to provide accurate predictions of system reliability. Due to the nature of the components involved in the models this does not seem feasible. Also recall that the definition of reliability being used (being within a certain error bound) is more strict than

the common definition of reliability. The aim of these models are to serve as a means of comparing different configurations and interactions (i.e. role of human operator, kinematic configurations, recovery modes, etc.) within the system. These comparisons can then be utilized in order to optimize desirable system characteristics.

The expanded model will be further developed and applied to the NGMH/ATD system in Chapter 4.

Chapter 4: Model Application

4.1 Introduction

This chapter describes the application of the reliability model to the Advanced Technology Demonstrator (ATD) for the Next Generation Munitions Handling System (NGMH). The NGMH/ATD system is currently being developed at Oak Ridge National Laboratory for the United States Air Force. It is being designed as a hydraulically powered teleoperated manipulator. The current design of the NGMH/ATD system is illustrated in Figure 4.1. The NGMH is intended to provide improved munitions handling capability for a variety of aircraft well into the next century. The goal for the NGMH system is to move objects weighing from 200lbs to 2600lbs while having a position resolution of less than 1 mm [5]. The ATD project is intended to serve as a test-bed for this new telerobotic technology. The ATD will support the development, testing, and comparative evaluation of emerging technologies that will eventually lead to the NGMH system.

Currently, the ATD system has eight degrees of freedom (see Appendix F for a schematic of joint variables). To achieve any desired position and orientation (pose) in three-dimensional space only six degrees of freedom are required. Therefore, one of the objectives for the ATD system is to identify which joints are most effective in the movement of the manipulator. The less effective redundant joints may be omitted in future designs. They may also be used to help with obstacle avoidance as well as to

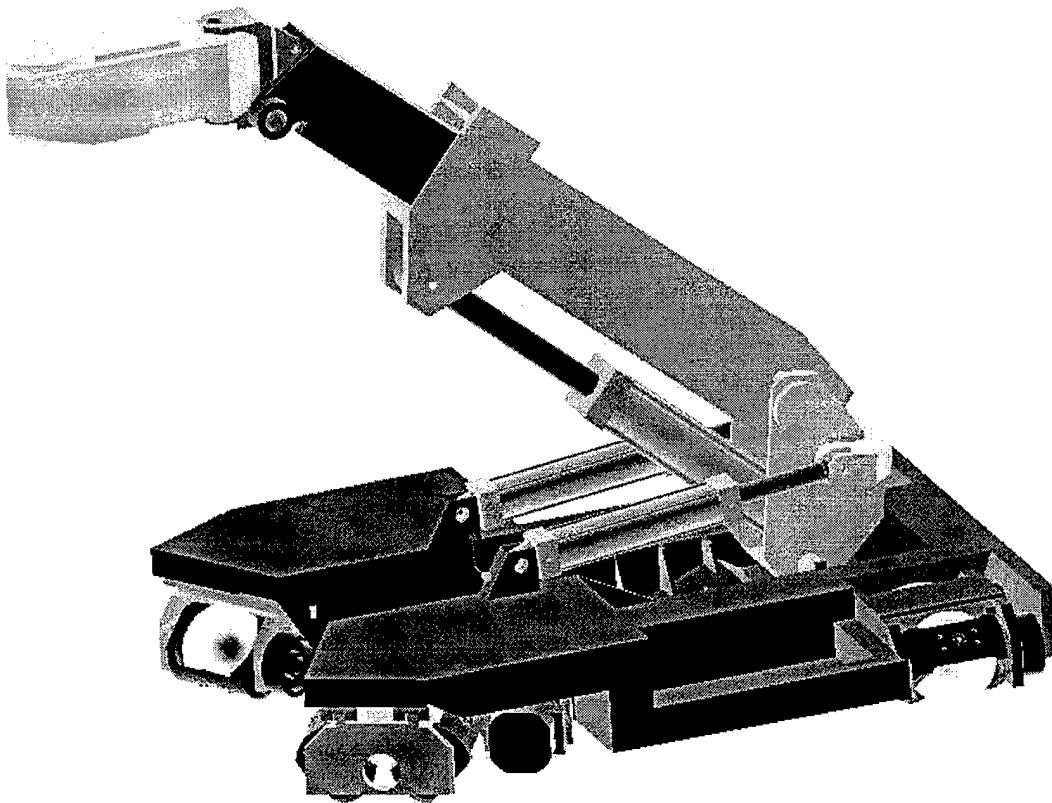


Figure 4.1 Illustration of Advanced Technology Demonstrator for the NGMH System

optimize certain aspects of system performance (i.e. minimize hydraulic fluid flow). In particular, joints 1 and 3 both provide similar types of motion for the manipulator. The effect of locking one these joints in various configurations and allowing the other to be used will be investigated later in this chapter.

The role of the human operator and the means of controlling the manipulator are another key item of interest in the development of the system. The current ATD design has a joystick device attached to the end-effector of the manipulator. There is a toggle switch on the joystick which toggles between positioning (first four joints) to orientation (last four joints). During the positioning phase of the manipulation, the end-effector is to

remain at a constant orientation. Once positioning has been accomplished, the toggle switch locks the first four joints and allows the same joystick to be used for the orientation of the manipulator. During both positioning and orientation, the manipulator follows the movements of the operator. Therefore, the operator's performance plays a key role in the overall effectiveness of the manipulator.

The NGMH/ATD system is being designed as a mobile system. The manipulator is attached to a movable platform. A patented wheel pod technology is being investigated to provide omnidirectional ground movement for the platform. This new technology would allow the platform to rotate and move in any direction simultaneously. Although the mobility of the system presents numerous design challenges, only the manipulator itself is investigated during the model application.

4.2 Model Application

In order to apply the reliability model developed in Chapter 3 to the ATD system, subtasks must first be defined and the four main components of a teleoperated robotic system must be carefully evaluated within each subtask. For the ATD system the overall task is assumed to involve the loading of a single munition onto a stationary rack. This task can be divided into a number subtasks including:

- Movement of the platform (driving)
- Gross positioning of manipulator (first four joints are operated while end-effector remains at constant orientation)
- Orientation of manipulator (last four joints operated while first four joints are locked)
- Attachment of munition
- Return to starting pose
- Loading of munitions onto manipulator

In order to simplify the model application, not all of these subtasks were included. For the purposes of this model application the subtasks were reduced to:

1. Positioning of manipulator
2. Orientation of manipulator
3. Attachment of munitions

These three subtasks can be best represented within the telerobotic reliability model developed previously. They also appear to be the most critical subtasks for successful completion of the overall task. From the definition of reliability being used, a successful task is one in which the end-effector of the manipulator is within a certain error bound in both its position and orientation [3]. In this particular application, this can be thought of in terms of the pose needed to successfully attach the munition to the munition rack. Having identified the three main subtasks to be analyzed, the four main components can now be evaluated within each subtask.

4.2.1 Human

Since the human component of the system must operate within three distinct subtasks, a failure rate was developed for each subtask. Ideally, these rates would be obtained using experimental data from the system under analysis. Since time constraints did not permit the collection of human error data on the actual ATD system, a previous study of human tracking error was used to provide a reasonable starting basis. Slight modifications are then introduced to account for some of the differences between the experimental data and the operator actions required for the ATD system.

In [26] Repperger *et al* present their study concerning pilot tracking error. Experimental data of human performance was collected to provide failure probabilities for use within Markov models. Two and three state Markov models were developed through a series of experiments with human subjects. The transition probabilities were determined between the respective states for each model. The two state model is of primary interest since it fits neatly within the semi-Markov model developed earlier in Chapter 3. Figure 4.2 depicts this simple two state model. State 1 represents a normal tracking operation and state 2 represents abnormal tracking.

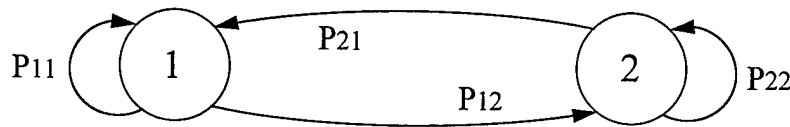


Figure 4.2 Two-State Human Markov Model

Through experimentation a steady state probability matrix was developed for this two state model. The steady state behavior of a good tracker is shown below.

$$\mathbf{P} = \begin{bmatrix} P11 & P12 \\ P21 & P22 \end{bmatrix} = \begin{bmatrix} 0.75 & 0.25 \\ 0.75 & 0.25 \end{bmatrix} \quad (4.1)$$

The tracking task that was investigated by Repperger *et al* is similar enough to the human tasks involved in the ATD system to provide useful insight, yet different enough that a few adjustments should be made. The pilot tracking task is considerably more difficult than the operator's actions for the ATD system. While an ATD operator may have a certain path or trajectory in mind, small deviations from this trajectory should not lead to

a failure. To account for this, scale factors, k , are introduced to the probability matrix (Equation 4.1) for each subtask. Using these scale factors, the probability of success can be determined by

$$P_{success} = k \cdot (P11) \quad (4.2)$$

and the probability of failure by

$$P_{failure} = 1 - k \cdot (P11) \quad (4.3)$$

These probabilities were then converted to failure rates by dividing by the time allotted for each experiment ($t=180$ seconds).

$$\lambda_h = \frac{P_{failure}}{t} \quad (4.4)$$

The scale factor chosen for each subtask is based on relative difficulty of the task. For subtask 1, positioning, the scale factor was chosen to be 1.2. A scale factor of 1.1 was used for the orientation subtask. The reasoning behind this decision was that orientation is more difficult than gross positioning, therefore, a lower probability of success is expected. Likewise, the scale factor for the attachment was chosen to be 1.3. One would expect this subtask to have the highest probability of success assuming the other subtasks were performed correctly. The scale factors and resulting failure rates are summarized in Table 4.1.

Table 4.1 Scale Factors and Human Failure Rates

Subtask	Scale Factor, k	$P_{success}$	Failure rate, λ_h (errors per sec)
1. Positioning	1.2	0.9	5.556E-4
2. Orientation	1.1	0.825	9.722E-4
3. Attachment	1.3	0.975	1.389E-4

The probability of recovery given a human error also becomes very important within the model. These recovery probabilities are also based largely on relative significance. For example, for the positioning subtask a recovery probability of 0.9 was assigned. This is due to the amount of time the operator has to recover from an error in the intended path or trajectory. Subtask 2 was also assigned a recovery probability of 0.9 for the same reasons. However subtask 3 was assigned a 0.75 probability of recovery due to the nature of the subtask. There is considerably less opportunity to recover from an error in the attachment of the munition than in the positioning and orientation of it. The recovery probabilities and human error rates are now available for use within a semi-Markov type simulation.

4.2.2 Hardware

In order to analyze the hardware reliability of the ATD system, engineering judgment was first used to determine the primary hardware components which affect system reliability. This process identified these critical components to be: hydraulics, electronics, bearings, engine, pump, force/torque sensors, and switches. Failure rates were estimated for each of these sub-systems using generic reliability data from the *Nonelectronic Parts Reliability Data* (NPRD) handbook and MIL-HDBK-217F [24,18]. When available, failure data pertaining to components used in a GM (ground mobile) environment was used. The failure rates determined for the hardware system are by no means an accurate prediction of the ATD's hardware reliability. The failure rate estimates generated often neglect components due either to a lack of reliability data or

incomplete system specifications. However, the failure rates do serve as reasonable estimates to be used within the reliability model.

The hydraulic system consists of servo-valves which are used for precision control of each actuator, hydraulic cylinders used to operate the first four joints, and hydraulic actuators used to operate joints within the wrist of the manipulator. The estimated failure rates for individual hydraulic components as well as the overall hydraulic failure rate are summarized in Table 4.2. Please note that the number of each component is placed in parentheses and that the failure rate shown is the failure rate estimate for a single component.

Table 4.2 Summary of Hydraulic Components Selected

Component Type	Failure rate (failures/ 10^6 hrs)
Servo-valve (7)	78.221
Hydraulic rotary actuator (1)	87.935
Hydraulic motor (1)	82.664
Hydraulic cylinder (4)	0.008
	718.1776

The electrical system for the ATD consists of processor cards, force/torque sensor cards, servo-cards, a VME backplane, and a number of cables and connectors. A basic parts count reliability model was attempted to develop a failure rate for the electrical system [18]. The components used in this analysis are summarized in Table 4.3. Obviously, this does not represent every component within the electrical system. Nonetheless, notice that the electrical system failure rate estimate is significantly lower than the estimate for the hydraulic system.

Table 4.3 Summary of Electrical Components Selected (λ s in failures/ 10^6 hours)

Component Type	Sub-components	λ for each sub-component	λ for each component type
030 Processor cards (2)	32 bit MOS microprocessor (2)	0.49	1.3
	256K EEPROM (2)	0.046	
	DRAM 1MB (4)	0.057	
force/torque sensor cards (3)	32 bit MOS microprocessor (1)	0.49	0.575
	256K EEPROM (1)	0.046	
	Voltage Regulator (1)	0.039	
MIL connectors (35)		0.11	0.11
			8.175

Since most joints are dependent on bearings to minimize joint friction, some of the key bearings were analyzed to determine their expected life. The basic rating for life of bearings is often represented as L_{10} life. L_{10} life is the number of revolutions that 90 percent of a group of bearings will complete or exceed before failure [31: 457]. This can be calculated using

$$L_{10} = \left(\frac{C}{F} \right)^a \quad (4.5)$$

where L_{10} is the basic life in millions of revolutions,
 $a = 10/3$ for roller bearings,
 C is the basic load rating provided by the manufacturer, and
 F is the actual load seen by the bearing.

This can be easily converted to a time to failure representation using

$$L_{10h} = \frac{(1.0 \times 10^6) L_{10}}{(60)n} \quad (4.6)$$

where L_{10h} is the basic life in hours and
 n is the rotational speed in rev/min.

Load data was obtained from simulations conducted at Oak Ridge on the first four joints. The simulation was conducted with the following assumptions: a 2600lb mass on the end-effector, d3 fully extended, θ_1 varied from 40° to 130° while θ_2 remained locked at -48° , and θ_4 varied to keep the end-effector horizontal. The largest magnitude force seen by each joint during the simulation was selected to be used in the bearing life calculation. The calculations for bearing life and failure rates are summarized in Table 4.4. The rotational speed, n , was assumed to be 60 rev/min. The C values were obtained from specifications given in the *SKF Bearing Product Catalog* [32] for the actual bearings used in the ATD design. Although only three joints were considered in the bearing calculation, this provides a reasonable estimate of bearing failure for use within the hardware reliability model.

Table 4.4 Summary of Bearing Life Calculations

Bearing Location	C (lbf)	max $ F $ (lbf)	L_{10h} (hrs)	λ (failures/ 10^6 hr)
Lower link - Base (2)	52,900	21,180	5872.1	170.3
Upper - Lower link (2)	44,800	17,005	7015.1	142.5
Wrist pitch (bottom)	18,600	7910	4802.7	208.2
Wrist pitch (top)	18,000	7910	4305.4	232.0
				1065.8

Other hardware components of interest include the force/torque sensors, the diesel engine, the toggle switches on the joystick, and the pump. These items are included in Table 4.5 along with the failure rate summaries from the hydraulic, electronic, and bearings. This table represents a summary of all hardware components considered within the model.

Table 4.5 Summary of Hardware Components

Component	Failure rate (failures/10 ⁶ hrs)
Hydraulic System	718.18
Electrical System	8.175
Bearings	1065.8
Pump	146.09
Diesel Engine	235.37
Switch (2)	4.8
Force/Torque Sensor (3)	80.0

In order to account for some of the wearout common to certain types of hardware components, a Weibull failure distribution was used for the hydraulic system, pump, and engine. An exponential failure distribution is assumed for the remaining hardware components. It may seem that the bearings should be included as a type of wearout component, however, the definition of bearing life used suggests that any form of wearout denotes failure. The overall hardware system is thus grouped into two categories: (1) components with an increasing failure rate (wearout) and (2) components with a constant failure rate. The components within each category are lumped together and represented by a single failure distribution.

The components having a constant failure rate can be represented by a single failure rate as follows

$$\begin{aligned}\lambda_d &= [8.125 + 2(4.8) + 3(80) + 1065.8] \\ \lambda_d &= 1325.5 \text{ failures}/10^6 \text{ hours}\end{aligned}\tag{4.7}$$

The components being represented by an increasing failure rate are first combined as

$$\begin{aligned}\lambda'_d &= [718.18 + 146.09 + 235.37] \\ \lambda'_d &= 1099.6 \text{ failurs}/10^6 \text{ hours}\end{aligned}\tag{4.8}$$

Assuming that this distribution possesses the same MTTF as the exponential, the Weibull parameters can be found using the relationship shown in Equation 3.20. If the shape parameter, β , is chosen to be 1.5, then the scale parameter, α , can be calculated as

$$\alpha = \left[\Gamma \left(1 + \frac{1}{\beta} \right) \right]^{-1} \cdot MTTF = 1.108 \left(\frac{1}{1099.6 \text{ failures} / 10^6 \text{ hrs}} \right) = 1007.6 \text{ hrs} \quad (4.9)$$

The decision to combine all wearout components into a single Weibull failure distribution was made in order to simplify the model application. These components could also have been modeled individually using different shape and scale parameters.

Since each subtask requires the use of all of the hardware components chosen, each subtask was assumed to possess the same failure distributions. These two failure distributions can now be used within a semi-Markov type simulation for each subtask.

4.2.3 Software

Since most software reliability models require extensive testing of software, a very general model was used to determine a reasonable software failure rate estimate for the ATD system. The model chosen is presented in Rome Laboratory's *Reliability Engineer's Toolkit* [25:124]. Using this model, the initial software failure rate, λ_o , is given by

$$\lambda_o = \frac{r_i K W_o}{I} \quad (4.10)$$

where r_i is the host processor speed (instructions/sec),
 K is the fault exposure ratio (default = $4.2E-7$),
 W_o is an estimate of the total number of faults in the initial program (default = 6 faults/1000 lines of code), and
 I is the number of object instructions (number of source lines of code times the expansion ratio)

The software used to control the ATD system is written in the C programming language. The expansion ratio given in [25] for C code is 2.5. Approximately 700 lines of code are contained within the program. The host processor speed is assumed to be 1 MIPS (million instructions per second). Using this information, the initial software failure rate can now be calculated as

$$\lambda_o = \frac{(1E6instr / sec)(4.2E - 7)\left(\frac{6faults}{1000lines}\right)(700lines)}{2.5(700)instr} \quad (4.11)$$

$$\lambda_o = .001008faults / sec$$

Assuming that a certain amount of time is allotted to debugging the initial program, a software reliability growth model can be used to account for improvements made in this development process. This model is given as

$$\lambda(t) = \lambda_o \exp[-\beta t] \quad (4.12)$$

where t is the time allotted for testing (in seconds) and
 β is the decrease in failure rate per failure occurrence given by

$$\beta = B \frac{\lambda_o}{W_o} \quad (4.13)$$

where B is the fault reduction factor (default = .955)

Assuming that 10 hours of testing time are used, the new software failure rate becomes

$$\beta = (.955) \left(\frac{.001008faults / sec}{4.2faults} \right) = 2.292E - 4 faults / sec$$

$$\lambda(36,000) = (.001008faults / sec) \exp[(2.292E - 4 faults / sec)(36,000sec)] \quad (4.14)$$

$$\lambda(36,000) = 2.6303E - 7 failure / sec$$

Obviously, the method used to obtain this estimate is very general in nature. However, it does result in a reasonable estimate of software reliability to be used in the applied model. Since each subtask requires the software to be working properly, the same software failure rate was used to define an exponential failure distribution for each subtask. This failure distribution can now be used within a semi-Markov type simulation.

4.2.4 Kinematic

The kinematic component of telerobotic reliability was evaluated through the use of the method introduced by Bhatti and Rao [3]. Each joint variable is assumed to be a random normal variable with a given mean and standard deviation associated with it. The mean is specified by the desired angle or length for each joint. The standard deviation is assumed to be equal to the control resolution for each joint. The control resolution of each joint can be estimated using [12:327]

$$\sigma = \frac{360^\circ}{2^n} \quad (4.15)$$

where n represents the number of bits of accuracy. The number of bits of accuracy and control resolution for each joint is summarized in Table 4.6.

Table 4.6 Control Resolution for each Joint

Joint Variable	Bits of Accuracy	Control Resolution, σ
θ_1	16	.00549°
θ_2	16	.00549°
d3	-	.001m
θ_4	12	.08789°
θ_5	12	.08789°
θ_6	12	.08789°
θ_7	12	.08789°

Notice that joint d3 is assumed to have a control resolution of 1mm. Kinematic reliability estimates were obtained through the use of FORTRAN simulations (see Appendix E).

These simulations produce point estimates of kinematic reliability for various kinematic configurations of the ATD. Since these estimates could not be directly converted to failure rates, the kinematic component of telerobotic reliability was not included in the semi-Markov type simulation used for the human, hardware, and software components.

For the positioning subtask Type III kinematic reliabilities were estimated. Of all positioning reliability types presented by Bhatti and Rao in [3], Type III is the most restrictive. The positioning of the manipulator must be within specified tolerances in all three-dimensions. The tolerances can be specified as maximum deviations from the mean values of the x , y , and z coordinates of the end-effector position. The tolerances used are $\pm 2\text{mm}$ for each coordinate. Type III kinematic reliability, $R_k(\text{III})$, for the ATD system can be represented as

$$R_k(x, y, z) = P \left\{ \begin{array}{l} (x_d - 0.002 \leq x \leq x_d + 0.002) \cup \\ (y_d - 0.002 \leq y \leq y_d + 0.002) \cup \\ (z_d - 0.002 \leq z \leq z_d + 0.002) \end{array} \right\} \quad (4.16)$$

Type V kinematic reliability is used for the orientation of the manipulator. This is the most restrictive orientational kinematic reliability presented by Bhatti and Rao. All three of the vectors representing the orientation of the end-effector must be completely specified. These vectors are commonly denoted as \mathbf{n} , \mathbf{s} , and \mathbf{a} and are called the normal, sliding, and approach vectors respectively. These vectors can be represented in terms of pitch, roll, and yaw angles (see Appendix G for this formulation). A tolerance of $\pm 0.2^\circ$ is

used for each of the orientation angles. Using these angles, Type V kinematic reliability, $R_k(V)$, for the ATD system can be represented as

$$R_k(\theta, \phi, \psi) = P \left\{ \begin{array}{l} (\theta_d - 0.2^\circ \leq \theta \leq \theta_d + 0.2^\circ) \cup \\ (\phi_d - 0.2^\circ \leq \phi \leq \phi_d + 0.2^\circ) \cup \\ (\psi_d - 0.2^\circ \leq \psi \leq \psi_d + 0.2^\circ) \end{array} \right\} \quad (4.17)$$

Since both position and orientation must be accurate to successfully load the munition, both Type III and Type V conditions must be satisfied to have a success within the orientation subtask. Total kinematic reliability, $R_k(\text{tot})$, represents the situation when both Equations 4.13 and 4.14 are satisfied. The FORTRAN simulation calculates $R_k(\text{III})$, $R_k(V)$, and $R_k(\text{tot})$ values for various configurations of the ATD system.

In order to apply the model developed in Chapter 3 to the ATD system arbitrary trajectories were selected to be analyzed. The positioning portion for three arbitrary trajectories is shown in Figure 4.3. Each trajectory begins in the upper-left portion of the plot and ends in the lower-right. The points along each trajectory represent joint configurations at which $R_k(\text{III})$ estimates were calculated. At the end of the positioning subtask, the first four joints are locked and orientation occurs. The orientation of the manipulator is assumed to be identical for all three trajectories chosen. Table 4.7 shows the sequence of joint angles used for the orientation of the manipulator. $R_k(\text{tot})$ estimates were calculated for each of these joint configurations. Appendix H contains further details of these three trajectories.

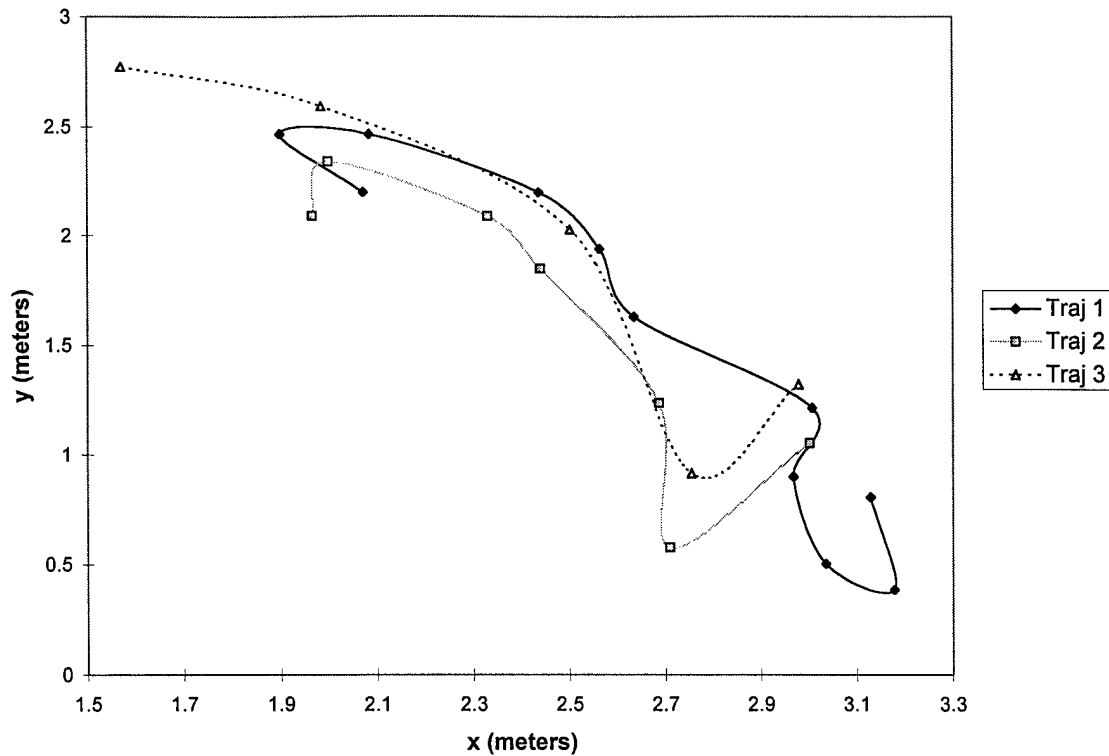


Figure 4.3 Trajectories for Positioning of Manipulator

Table 4.7 Orientation used for all Trajectories

θ_5	θ_6	θ_7
-10°	0°	90°
-20°	0°	90°
-20°	20°	90°
-20°	20°	100°
-20°	40°	100°

Kinematic reliability values were calculated at points along each trajectory. The minimum kinematic reliability value for each subtask is chosen to serve as a point estimate to represent that subtask. Choosing the minimum R_k values corresponds to the study conducted by Schneider [29]. The minimum kinematic reliability values for positioning and orientation are summarized in Table 4.8.

Table 4.8 Minimum Kinematic Reliability Values

Trajectory	$R_k(\text{III})$ Positioning	$R_k(\text{tot})$ Orientation
1	0.85	0.74
2	0.866	0.744
3	0.866	0.74

The final subtask, attachment of the munition, was assumed to have a kinematic reliability equal to 1.0. Since the overall model requires each subtask to be completed successfully (refer to Equation 3.8), errors in kinematic reliability should appear within the first two subtasks.

4.3 Results

The results obtained from the application of telerobotic reliability model to the ATD system are divided into two main sections. The first set of results is obtained from a task analysis of the three trajectories shown in Figure 4.3. This analysis includes a semi-Markov simulation for the human, hardware, and software components as well as a FORTRAN simulation to obtain kinematic reliability estimates. A reliability index, R_I , is developed for ten replications of the overall task of loading a munition onto a stationary rack. This reliability index is not intended to be a prediction of actual reliability, rather to serve as a means of comparing different configurations of the system.

The second set of results is obtained regarding only the kinematic reliability estimates of the ATD system. These results are obtained through simulations of various kinematic configurations over the range of the joint variables.

4.3.1 Task Analysis

Since no typical trajectory has been specified for the ATD, the arbitrary trajectories of Figure 4.3 are selected for analysis. Each trajectory was analyzed for ten repetitions of the overall task of loading a munition onto a stationary rack. The first subtask was assumed to take an average of 30 seconds, the second 20 seconds, and the third 10 seconds. The main parameters used within the semi-Markov simulation for the human, hardware, and software components are summarized in Table 4.9. The results from the semi-Markov simulation are shown in Table 4.10. Appendix B contains the SLAM II details of this simulation.

Table 4.9 Parameters for Semi-Markov Simulation (λ s in failures per second)

Subtask	Human λ_h	Hardware		Software λ_s	Total Time (sec)
		Exp. λ_d	Weibull, α ($\beta=1.5$)		
1	5.556E-4	3.682E-7	3.627E6 sec	2.630E-7	300
2	9.722E-4	3.682E-7	3.627E6 sec	2.630E-7	200
3	1.389E-4	3.682E-7	3.627E6 sec	2.630E-7	100

Table 4.10 Summary of Semi-Markov Simulation Results

Subtask, i	Human Recovery	Human Failure	Hardware Failure (Exp.)	Hardware Failure (Weibull)	Software Recovery	Software Failure	$R(T_i)$
1	448	39	0	2	0	0	0.959
2	521	46	0	0	0	0	0.954
3	16	9	0	0	0	0	0.991

From the semi-Markov simulation, the human appears to be the critical component within this telerobotic system. For subtask 1, 95.1% of all system failures are due to the human operator. In subtasks 2 and 3, the operator is responsible for 100% of the system failures. Since the human, hardware, and software components are assumed to behave the same for

any given trajectory, the results from this semi-Markov simulation are used for all three trajectories in the development of the reliability index.

Once the semi-Markov simulation results were obtained, these reliability values are combined with R_k values (from Table 4.8) to obtain a reliability index measure for each subtask. Equation 3.22 is utilized to obtain R_I values for each subtask and then Equation 3.23 is applied to calculate an overall reliability index value. Table 4.11 displays the results of these calculations.

Table 4.11 Reliability Index Values

Subtask	Trajectory 1 R_{Ii}	Trajectory 2 R_{Ii}	Trajectory 3 R_{Ii}
1	0.81515	0.830494	0.830494
2	0.70596	0.709776	0.70596
3	0.991	0.991	0.991
Overall	0.5703	0.5842	0.5810

Recall that these reliability index values are for comparative evaluation only.

From this brief analysis, the second trajectory results in the highest reliability index value.

Therefore, it provides the highest measure of reliability compared to the other two trajectories analyzed. Within all three trajectories, subtask 2 appears to be the most critical subtask. This is a result of the kinematic reliability estimates obtained for the orientation of the manipulator. Since both Type III and Type V kinematic reliability specifications must be satisfied during the orientation subtask, it has considerably lower R_k values associated with it than the other subtasks.

4.3.2 Kinematic Analysis

The kinematic analysis of the ATD system is performed through the use of a FORTRAN simulation over range of joint variables (refer to Appendix E). The analysis presented here pertains only to the kinematic reliability of the system. The other three components are not included. Two basic kinematic studies are conducted. The first study takes place from the point of view of the workspace of the manipulator. It deals with the positioning subtask only. Type III kinematic reliability estimates are obtained for a number of positions within the workspace of the manipulator. The second study involves looking at kinematic reliability variations over the entire range of joint variable configurations. Both the positioning and orientation subtasks are investigated in this manner.

The positioning workspace used for the kinematic reliability calculations is shown in Figure 4.4. This workspace represents the x and y coordinates (assuming feasible y values must be positive) obtained by varying the joint variables as follows

$$\begin{array}{rclclcl} 40.0^{\circ} & \leq & \theta 1 & \leq & 110.0^{\circ} \\ -124.0^{\circ} & \leq & \theta 2 & \leq & -24.0^{\circ} \\ 74.56 \text{ in} & \leq & d3 & \leq & 98.56 \text{ in} \\ -79.9^{\circ} & \leq & \theta 4 & \leq & 10.1^{\circ} \\ & & \theta 5 = \theta 6 = \theta 7 = 0^{\circ} & & \end{array}$$

where $\theta 1$ is varied by 10° increments, $\theta 2$ is varied by -20° increments, and finally $d3$ is varied by 6 inch increments. Although the workspace has been constrained to remain within the x-y plane for this analysis, a three-dimensional workspace analysis may be investigated in a similar manner.

In order to keep the end effector horizontal, θ_4 is determined using the constraint

$$\theta_4 = -(\theta_1 + \theta_2) \quad (4.18)$$

where feasible within θ_4 's range. If this value falls outside of θ_4 's prescribed operating range, then the nearest limit is used.

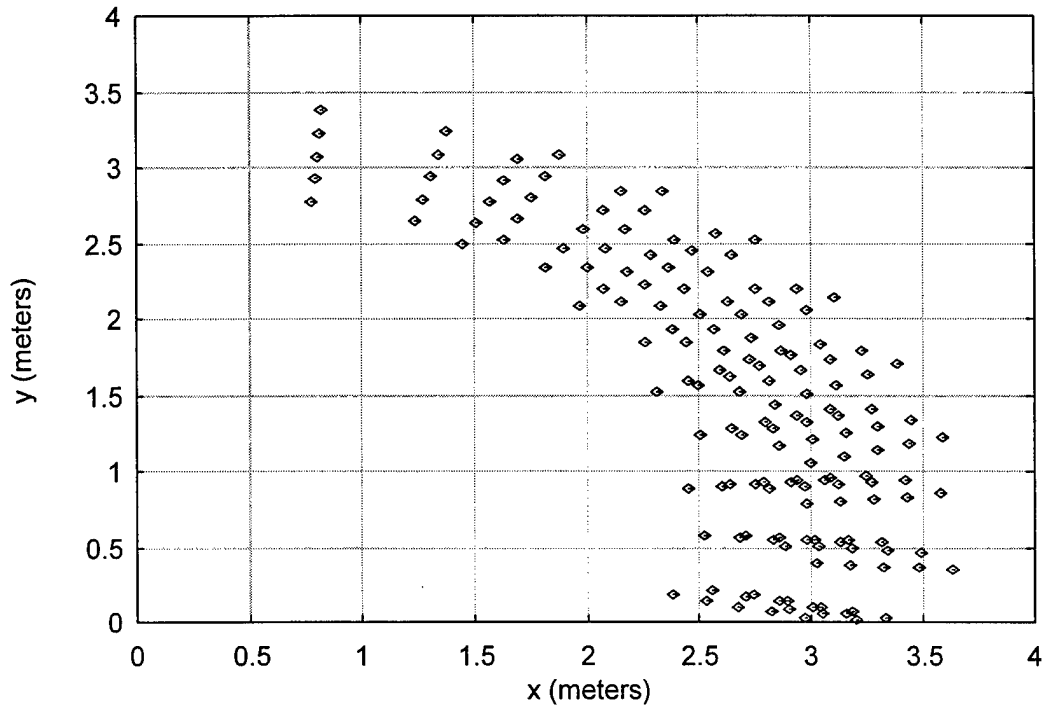


Figure 4.4 X-Y Coordinates over Positioning Workspace

Through the use of FORTRAN simulations, Type III kinematic reliability estimates are obtained for each point depicted in the workspace. The length d_3 is assumed to be exact, while all other joint variables are sampled from a random normal distribution. These results are illustrated in both Figures 4.5 and 4.6. While there does not appear to be a definite trend or pattern in Figure 4.5, this three-dimensional plot does give a good indication of the variance for the $R_k(\text{III})$ values obtained. The $R_k(\text{III})$ values

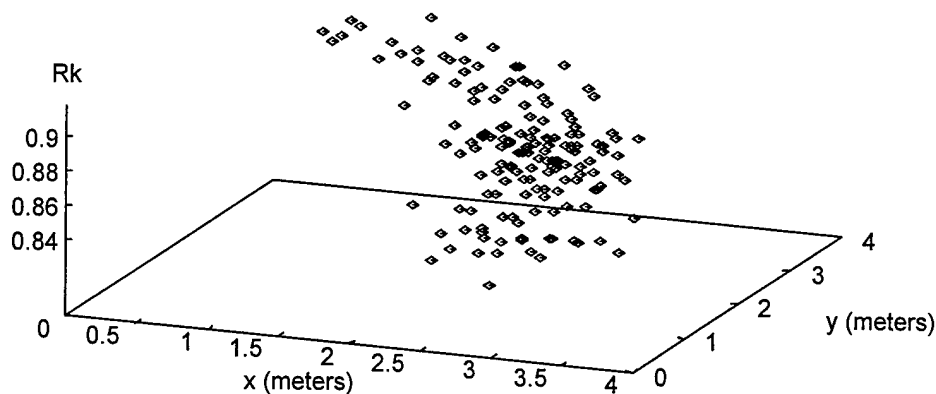


Figure 4.5 Type III Kinematic Reliabilities over Positioning Workspace (3-D Plot)

estimated were between 0.836 and 0.918. Therefore, even over a simple x-y constrained positioning workspace, the measure of kinematic reliability is highly variable.

An alternate illustration of the Type III kinematic reliability estimates is given in Figure 4.6. This figure provides additional insight into the $R_k(\text{III})$ point estimates. The dark gray points correspond to $R_k(\text{III})$ values greater than 0.90, medium gray points are greater than 0.875, and light gray points have values greater than 0.85. Notice that a few of the points, with values less than 0.85 do not appear on this plot. While this figure provides a clearer picture of the kinematic reliability estimates, it is still difficult to see any definite trends or patterns in the values.

The second kinematic study takes place from the point of view of the joint angle configurations. This study is concerned with both the positioning and orientation of the manipulator. For the positioning of the manipulator, the same joint ranges are used in

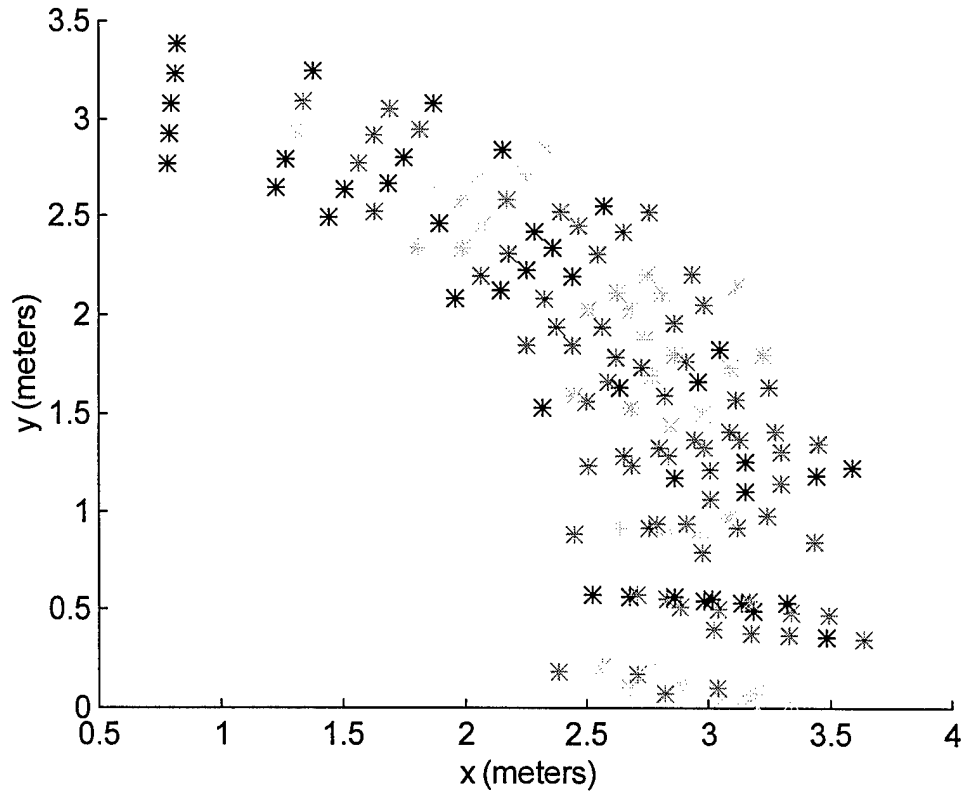


Figure 4.6 Type III Kinematic Reliabilities over Positioning Workspace

this study as in the x and y study. In fact, the same data used to construct Figures 4.5 and 4.6 is now used to develop Figure 4.7. However, for this study, negative y values are also included. This was done to maintain continuity of the plot. Also, depending on the exact location of the base reference frame (determined by the height of the platform), some negative y values may be permissible. Figure 4.7 depicts the variation in $R_k(\text{III})$ values when d_3 is assumed to be exact. The data shown in Figure 4.7 is obtained over each interval of d_3 by varying θ_2 over its permissible range using -20° increments. For each value of θ_2 , θ_1 is then varied over its permissible range using 10° increments.

There appears to be a slight decrease in the $R_k(\text{III})$ values as d_3 is increased. This is expected since a longer length should cause the variations in the first two joint angles to result in larger deviations from the desired position. Also, for each specified length of d_3 , there appears to be a similar pattern in the $R_k(\text{III})$ values. Each interval of d_3 seems to have a “spike” in the $R_k(\text{III})$ estimates towards the end of the interval. These “spikes” correspond to configurations at which $\theta_2 = -124^\circ$ and $\theta_1 = 50^\circ$.

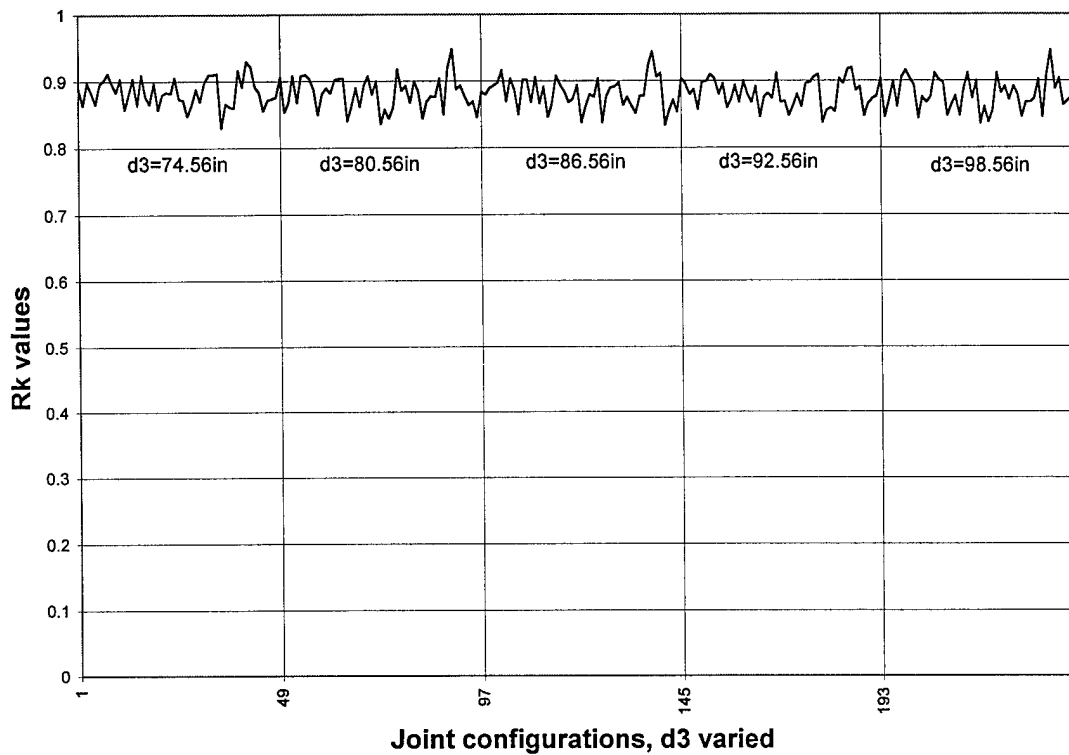


Figure 4.7 Kinematic Reliability Values for Joint Configurations (d_3 Varied)

For the orientation of the manipulator, an arbitrary position is assumed to be selected. Recall that the orientation of the manipulator involves only the last three joint angles. Thus, the first four joints are locked in place and the last three joint angles are varied as follows

$$\begin{aligned}
 -50.0^\circ &\leq \theta_5 \leq 50.0^\circ \\
 -80.0^\circ &\leq \theta_6 \leq 80.0^\circ \\
 80.0^\circ &\leq \theta_7 \leq 100.0^\circ \\
 \theta_1=50^\circ, \theta_2=-44^\circ, d_3=80.56 \text{ in}, \theta_4=-6^\circ
 \end{aligned}$$

where θ_7 is varied by 10° increments, θ_6 by -20° increments, and finally θ_5 by -10° increments. The length d_3 was assumed to be exact, while all other joint variables were sampled from a random normal distribution. Through the use of a FORTRAN simulation the data was obtained to create Figure 4.8. The data shown in Figure 4.8 is obtained over each interval of θ_5 by varying θ_6 over its permissible range. For each value of θ_6 , θ_7 is then varied over its permissible range. Notice that all three R_k values are now calculated.

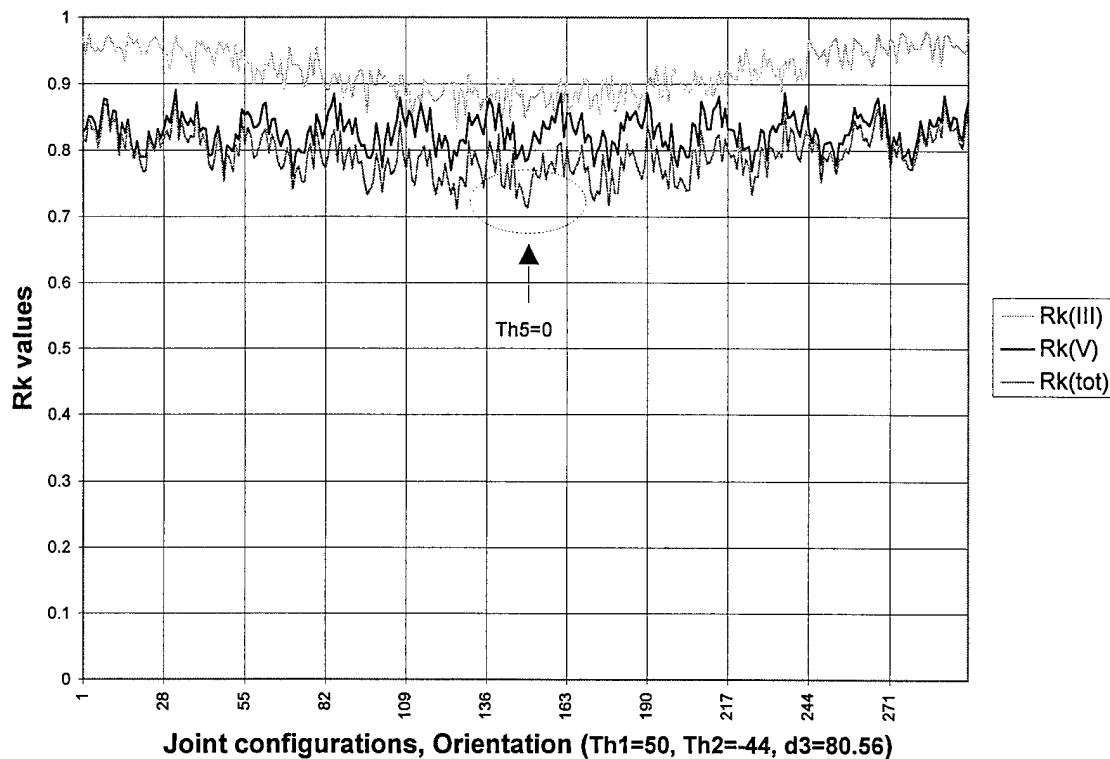


Figure 4.8 Kinematic Reliability Values for Joint Configurations (Orientation)

The important measure for the orientation subtask is the $R_k(\text{tot})$ value, since both position and orientation must be within specified tolerances in order for a success to be counted within the simulation.

Notice that the Type III kinematic reliability estimates are significantly higher than the Type V estimates. Also, the Type V estimates tend to have a higher magnitude of variability associated with them. The “spikes” seen within the $R_k(V)$ estimates correspond to configurations where θ_6 is between 40° and 60° . The $R_k(\text{tot})$ estimates seem to correspond fairly well with the Type V estimates in addition to showing a similar overall shape to the Type III values. All three of the kinematic reliability estimates shown appear to be symmetric about the center of the plot.

The arrow on the above plot is directed toward the middle section of the plot where $\theta_5=0^\circ$. This area seems to result in the lowest $R_k(\text{tot})$ estimates. As the θ_5 angle is rotated further away from 0° , the $R_k(\text{tot})$ estimates improve. This makes sense because when θ_5 is at or near 0° the manipulator has a larger radius of operation. Generally speaking, points further out in the workspace have a lower kinematic reliability.

Throughout this kinematic analysis, joint d3 was assumed to be exact. Appendix C contains plots related to a more complete kinematic analysis over the entire range of joint variables. Both position and orientation are included on these plots. Also, Appendix D contains similar plots for variations over the entire range of joint variables when d3 is sampled from a random normal distribution and θ_1 is assumed to be exact.

4.4 Optimization Issues

As stated earlier, the reliability index developed for task analysis can be used for comparative evaluation of system performance. This reliability index can be used to determine the best configuration of a telerobotic system subject to a number of constraints and restrictions. The configuration of a telerobotic system can be attributed to a number of factors including: role of human operator, the trajectories chosen for positioning and orientation, the kinematic configuration of the manipulator along these trajectories (with redundant joints), the various hardware components used, the software used to operate the system, etc. This problem can be expressed as follows

$$\begin{aligned} \text{Max } f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) &= R_I \\ \text{subject to } g(\mathbf{x}_i) &\leq y_i \mathbf{b}_i \quad i=1, \dots, n \\ y_1 + y_2 + \dots + y_n &= 1 \end{aligned} \tag{4.19}$$

where $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ represents vectors of design variables for the various system configurations,
 $g(\mathbf{x}_i) \leq \mathbf{b}_i$ represents the constraints associated with the i th configuration,
 y_1, y_2, \dots, y_n are binary variables (i.e. either 0 or 1), and
 n is the number of configurations under consideration.

The solution to this problem involves optimizing each configuration and then choosing the configuration that provides the highest reliability index. This can be done for each subtask of the system or generalized to include the overall task of the system. Although this problem has been written as a mathematical program, a complete enumeration of possible system configurations is probably the most practical solution technique.

The main decision variables within this type of analysis will be the role of the human operator within the system and the kinematic configurations and trajectories of the manipulator. These system specifications have not only been shown to have the highest

impact on the reliability index, but are probably the easiest parameters to change within a teleoperated robotic system.

4.5 Summary

The reliability model developed in Chapter 3 has been applied to the NGMH/ATD system to generate reliability indexes for three arbitrary trajectories. These indexes provide a relative measure of reliability that can be used in a comparative evaluation of a teleoperated robotic system. A purely kinematic analysis has also been presented for the kinematic component of the system. Although this model application was performed using limited system specifications and generalized reliability data, some important insights have been gained. The first of which is that the human component of the model is of primary importance. Secondly, the kinematic configuration of the manipulator has a large impact on the reliability measure. This model application can be further extended in order to optimize system reliability by comparing feasible configurations of the system.

Chapter 5. Conclusions and Recommendations

5.1 Summary of Research

This research effort has led to a reliability index model which can be applied to any teleoperated robotic system. The reliability model includes the following four main components: (1) human, (2) software, (3) hardware, and (4) kinematic. Each of these components are essential ingredients within a telerobotic system. Based on an extensive literature review, no other complete telerobotic reliability models have been found. While attempts have been made to model various combinations of these four components, no model has been designed to include all four. Therefore, this research effort has led to a unique telerobotic reliability model.

The reliability index is developed through the use of two underlying models. A semi-Markov model is utilized for the human, software, and hardware components, while a separate simulation program is applied for the kinematic component. Subtasks within the overall task of the system are then analyzed using these two models. The reliability index model provides a relative measure of reliability that can be used to compare various feasible configurations of a telerobotic system. These comparisons can be systematically performed in order to optimize system performance.

The reliability index model has been successfully applied to the Advanced Technology Demonstrator (ATD) which is being developed at Oak Ridge National Laboratory as a predecessor for the Next Generation Munitions Handler (NGMH). Although limited system specifications and generic reliability data were used, the

application of this model has resulted in a number of useful insights into the operation of the NGMH/ATD system. The human operator was found to be one of the key components out of the four components considered. This is no surprise since the movement of the manipulator is directly dependent on the human operator. Also, the kinematic configurations of the manipulator were found to have a large impact on the reliability measure obtained.

From the three subtasks considered in the model application, the orientation subtask was found to be the "weak link" in the overall task of the manipulator. This is due to the fact that both the position and orientation of the end-effector are critical within the orientation subtask. Therefore, the kinematic reliability estimates obtained within this subtask are generally much lower than those for the other subtasks. Also, the human operator was assumed to have the highest probability of making an error within the orientation subtask. Of course, these results are based on a cursory analysis of the NGMH/ATD system. A good deal of research remains in order to refine and better apply the reliability index model.

5.2 Recommendations

Although the model developed here provides a good comparative measure of telerobotic reliability, this research should be extended to further develop the model. A number of factors should be investigated for continued development and use of the model. These factors relate to both the reliability model in general as well as its application to the NGMH/ATD system.

5.2.1 Human Component

As shown in the reliability model application, the human component represents one of the key factors within a telerobotic system. Recall that the method used to construct human error rates for the model application was based on a previous study on pilot tracking error in [26]. Experimental data collected during the pilot tracking study was used as a starting basis for determining human reliability within the ATD system. A more accurate human model should be developed through experimentation with the actual ATD system. The data collected through this experimentation should be provide error rate estimates for the human operator. This human reliability data should be highly dependent on the role of the human operator (i.e. specific actions taken by the operator) and the ability of the operator to recover from an error.

5.2.2 Kinematic Component

A more detailed analysis of the kinematic component could also be undertaken to provide further insight into the ATD system. Recall that the control resolution of each joint was assumed to represent the standard deviation of each joint angle. Joint d3 was simply assumed to have a standard deviation of 1mm. Experimentation on the manipulator could be conducted in order to obtain a better estimate of these joint variations. Data could be collected to determine the actual standard deviation of each joint. These values could then be inserted into the kinematic reliability simulation program.

The order in which joint variables are varied within the FORTRAN simulation is another area that can be investigated. Consider the plots of kinematic reliability estimates for various configurations of the manipulator as shown in Figures 4.7 and 4.8. By altering the order in which these joint configurations are presented, similar plots may be obtained. These new plots may show additional trends and patterns that are not currently apparent.

Using the minimum R_k values in the reliability index calculation may not provide the best representation of the kinematic reliability throughout a subtask. Development of a technique to express the kinematic reliability in terms of the time domain would be beneficial. This would not only provide a better description of the kinematic reliability of the system, but would allow the kinematic component to be included within a semi-Markov model as well.

The kinematic reliability analysis is concerned only with kinematic configuration of the manipulator along with variations in joint accuracy. The forces and torques seen by each joint are not included in this analysis. Bhatti has introduced a measure of dynamic reliability to account for these interactions. He defines dynamic reliability as the probability that the end-effector location and/or velocity will not exceed the prescribed tolerances along a given trajectory [2]. The actual analysis of dynamic reliability becomes very complex due to the nonlinearity of the equations involved. However, this analysis is necessary to probabilistically quantify the force operation of the NGMH/ATD system. A successful application of this dynamic reliability measure to the ATD system may result in a number of additional insights.

Another recommendation concerning the kinematic component of the system, is to utilize the inverse kinematic equations. In the present kinematic analysis, joint variables must first be specified to calculate the position and orientation of the end-effector. The inverse kinematic equations would allow the joint variables to be determined for a given position and orientation of the end-effector.

5.2.3 Hardware and Software Components

A more detailed model of hardware and software components may provide additional insight into the system. Recall that generic reliability data was used to generate the parameters used within the hardware failure distributions for the model application. Also, the wearout components were lumped together using a single Weibull failure distribution. These wearout components could also be modeled separately using distinct failure distributions. Redundancy of hardware components could also be investigated. However, the usefulness of these efforts is questionable. The reliability index model has clearly shown that the human and kinematic components are the most important within the model. As long as the hardware and software models provide reasonable estimates, they should be sufficient for the reliability index development. In addition, the hardware and software components of a telerobotic system are the least flexible components in terms of altering the system configuration.

5.2.4 Fault Tolerance

The use of fault tolerance within a teleoperated robotic system should also be further investigated. In the reliability index model presented here, recovery from human

and software errors is represented by a probability of recovery times a recovery rate. This simple representation does not take into account the specific modes of failure that may be experienced by the system. A detailed fault-tree analysis analysis can be conducted to identify these failure modes. Once these failure modes have been determined, conditional probabilities of recovery may be developed for a given failure mode. Also, the recovery rates may be modified to account for the different modes of failure. In this manner, a more descriptive representation of the system's fault tolerance can be achieved.

5.2.5 Continuous Model

The use of a continuous semi-Markov model may also be investigated. Using a continuous system model would allow the reliability analysis to be applied to a much longer time period. In order for a continuous model to be developed, repair or recovery rates would have to be specified for all components within the model. A continuous model would be of particular interest if the kinematic component of the system could be represented in the time domain. A system availability index could then be developed as a comparative measure of system performance.

5.2.6 Trajectory Planning

The results of detailed human experimentation could be used in conjunction with a more complete kinematic reliability analysis to develop a set of recommended trajectories. While the final design of the NGMH system is intended to provide a high degree freedom and flexibility to the human operator, these recommended trajectories may be beneficial for training purposes.

5.3 Conclusion

A reliability index model has been developed for a general teleoperated robotic system. This reliability model includes the human, software, hardware, and kinematic components of a telerobotic system. The reliability index model developed through this research effort could have a number of practical uses. Used correctly, this could be an important design tool for any telerobotic system. This comparative reliability measure could be used to optimize system performance by selecting optimal trajectories of the manipulator and adjusting the role of the human operator. It could also be used to optimize the use of redundant joints of the manipulator for a given trajectory. Key components and subtasks within a telerobotic system may be identified as well.

While this research effort has laid an important foundation for a reliability analysis of a teleoperated robotic system, a number of additional research opportunities have also been identified. This research has developed a basic framework from which to begin these further studies.

Appendix A. First-Order Markov Model Solution

Analytical Results (from MathCAD)

$$\begin{aligned} \text{variables :} \quad Lr &:= 36 & Lh &:= \frac{1}{200} & Ls &:= \frac{1}{1500} & Ld &:= \frac{1}{2000} & Lk &:= \frac{1}{1000} \\ & & Phr &:= .5 & Psr &:= .5 & & & & \end{aligned}$$

$$\begin{aligned} \text{denominator of fo(s) :} \\ &= (s + Lr) \cdot (s + Lh + Ls + Ld + Lk) - Lr \cdot (Phr \cdot Lh + Psr \cdot Ls) \\ &= s^2 + s \cdot Lh + s \cdot Ls + s \cdot Ld + s \cdot Lk + Lr \cdot s + Lr \cdot Lh + Lr \cdot Ls + Lr \cdot Ld + Lr \cdot Lk - Lr \cdot Phr \cdot Lh - Lr \cdot Psr \cdot Ls = 0 \end{aligned}$$

solving for the roots of s : where $(s-v_0)(s-v_1)=0$

$$v := \begin{bmatrix} -\frac{1}{2} \cdot Lh - \frac{1}{2} \cdot Ls - \frac{1}{2} \cdot Ld - \frac{1}{2} \cdot Lk - \frac{1}{2} \cdot Lr + \frac{1}{2} \cdot \sqrt{Lh^2 + 2 \cdot Lh \cdot Ls + 2 \cdot Lh \cdot Ld + 2 \cdot Lh \cdot Lk + Ls^2 \dots} \\ -\frac{1}{2} \cdot Lh - \frac{1}{2} \cdot Ls - \frac{1}{2} \cdot Ld - \frac{1}{2} \cdot Lk - \frac{1}{2} \cdot Lr - \frac{1}{2} \cdot \sqrt{Lh^2 + 2 \cdot Lh \cdot Ls + 2 \cdot Lh \cdot Ld + 2 \cdot Lh \cdot Lk + Ls^2 \dots} \end{bmatrix}$$

$$v = \begin{pmatrix} -0.0043329923 \\ -36.0028336744 \end{pmatrix}$$

$$A1 := \frac{v_0 + Lr}{(v_0 - v_1)}$$

$$A2 := \frac{v_1 + Lr}{(v_1 - v_0)}$$

$$A1 = 0.9999212835$$

$$A2 = 7.871645605310^{-5}$$

$$R(t) := A1 \cdot \exp(v_0 \cdot t) + A2 \cdot \exp(v_1 \cdot t)$$

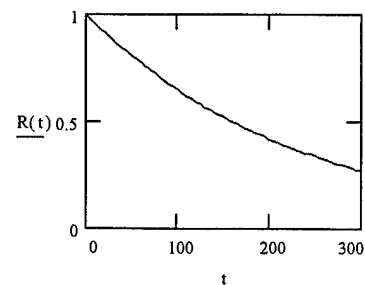
$$R(0) = 1 \quad R(5) = 0.978 \quad R(15) = 0.937 \quad R(50) = 0.805 \quad R(100) = 0.648$$

$$MTTF := \int_0^{10000} A1 \cdot \exp(v_0 \cdot t) + A2 \cdot \exp(v_1 \cdot t) \, dt$$

$$MTTF = 230.77$$

$$MTTF = \frac{2}{(2 \cdot Ld + 2 \cdot Lk + Lh + Ls)} = 230.769$$

$$t := 0, 5 \dots 300$$



SLAM II Network Statements

```
;
  CREATE,,,1,,1;
  ACTIVITY;
  ASSIGN,ATrib(2)=EXPON(200,4),ATrib(3)=EXPON(1500,3),ATrib(4)=EXPON(2000,
  7),ATrib(5)=EXPON(1000,5);
  ACTIVITY;
OK  GOON,4;
  ACTIVITY;
  ACTIVITY,,,HREC;
  ACTIVITY/3,ATrib(4),,HARD;
  ACTIVITY/4,ATrib(5),,KINE;
SREC GOON,1;
  ACTIVITY/1,ATrib(2);
HUM  GOON,1;
  ACTIVITY/6,,0.5;
  ACTIVITY,,0.5,ZAAC;
  COLCT,XX(6),HUMAN RECOVERY,,1;
  ACTIVITY,,,SREC;
ZAAC COLCT,XX(2),HUMAN ERROR,,1;
  ACTIVITY;
FAIL GOON,1;
  ACTIVITY;
  COLCT,XX(1),FAILURES;
  ACTIVITY;
  TERMINATE,1;
HREC GOON,1;
  ACTIVITY/2,ATrib(3),,SOFT;
SOFT GOON,1;
  ACTIVITY/5,,0.5;
  ACTIVITY,,0.5,ZAAD;
  COLCT,XX(7),SOFT RECOVERY,,1;
  ACTIVITY,,,HREC;
ZAAD COLCT,XX(3),SOFTWARE ERROR,,1;
  ACTIVITY,,,FAIL;
HARD GOON,1;
  ACTIVITY;
  COLCT,XX(4),HARDWARE ERROR,,1;
  ACTIVITY,,,FAIL;
KINE GOON,1;
  ACTIVITY;
  COLCT,XX(5),KINEMATIC ERROR,,1;
  ACTIVITY,,,FAIL;
  END;
```

SLAM II Control Statements

```
GEN,MATT RUMMER,THESIS,1/29/1997,5000,N,N,Y/Y,N,Y/S,132;
LIMITS,1,6,6;
NETWORK;
INITIALIZE,,15,N;
FIN;
```


Output Report

DATE **/**/1997

RUN NUMBER 5000 OF 5000

CURRENT TIME .1500E+02

STATISTICAL ARRAYS CLEARED AT TIME .0000E+00

STATISTICS FOR VARIABLES BASED ON OBSERVATION

	MEAN VALUE	STAND. DEV.	COEFF. OF VARIATION	MIN VALUE	MAX VALUE	NO.OF OBS
HUM RECOVERY	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	254
HUMAN ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	265
FAILURES	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	421
SOFT RECOVERY	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	27
SOFTWARE ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	31
HARDWARE ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	47
KINEMATIC ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	78

FILE STATISTICS

FILE NUMBER	LABEL/TYPE	AVERAGE LENGTH	STANDARD DEVIATION	MAXIMUM LENGTH	CURRENT LENGTH	AVERAGE WAIT TIME
1		.000	.000	0	0	.000
2	CALENDAR	4.000	.000	4	4	6.667

REGULAR ACTIVITY STATISTICS

ACTIVITY INDEX/LABEL	AVERAGE UTILIZATION	STANDARD DEVIATION	MAX UTIL	CURRENT UTIL	ENTITY COUNT
1	1.0000	.0000	1	1	0
2	1.0000	.0000	1	1	0
3	1.0000	.0000	1	1	0
4	1.0000	.0000	1	1	0
5	.0000	.0000	1	0	0
6	.0000	.0000	1	0	0

Appendix B. SLAM II Program and Output for ATD System

The Network Statements, Control Statements, and Network Diagram shown all correspond to the SLAM II program used for the semi-Markov simulation of Subtask 1. The simulations for the other two subtasks require only slight modifications to this program. These modifications include: the human failure rate (ATRIB(2)), the time of creation for the first entity (create node), and the start and end times for the simulation (control statements). The output results are given for all three subtasks evaluated for the ATD system.

Network Statements

```
;
  CREATE,,,1,,1;
  ACTIVITY;
  ASSIGN,ATRIB(5)=WEIBL(3627378,1.5,7),ATRIB(2)=EXPON(1800,4),ATRIB(3)=
  EXPON(3801848,3),ATRIB(4)=EXPON(2711276,6);
  ACTIVITY;
OK  GOON,4;
  ACTIVITY;
  ACTIVITY,,,HREC;
  ACTIVITY/3,ATRIB(4),,HARD;
  ACTIVITY/4,ATRIB(5),,HYD;
SREC GOON,1;
  ACTIVITY/1,ATRIB(2);
HUM  GOON,1;
  ACTIVITY/6,,0.9;
  ACTIVITY,,0.1,ZAAB;
  COLCT,XX(6),HUMAN RECOVERY,,1;
  ACTIVITY,,,SREC;
ZAAB COLCT,XX(2),HUMAN ERROR,,1;
  ACTIVITY;
FAIL GOON,1;
  ACTIVITY;
  COLCT,XX(1),FAILURES;
  ACTIVITY;
  TERMINATE,1;
HREC GOON,1;
  ACTIVITY/2,ATRIB(3),,SOFT;
SOFT GOON,1;
  ACTIVITY/5,,0.5;
  ACTIVITY,,0.5,ZAAC;
```


COLCT,XX(7),SOFT RECOVERY,,1;
ACTIVITY,,,HREC;
ZAAC COLCT,XX(3),SOFTWARE ERROR,,1;
ACTIVITY,,,FAIL;
HARD GOON,1;
ACTIVITY;
COLCT,XX(4),HARDWARE ERROR,,1;
ACTIVITY,,,FAIL;
HYD GOON,1;
ACTIVITY;
COLCT,XX(5),HYDRAULICS,,1;
ACTIVITY,,,FAIL;
END;

Control Statements

GEN,MATT RUMMER,THESIS,**/**/1997,1000,N,N,Y/Y,N,Y/S,132;
LIMITS,1,6,6;
NETWORK;
INITIALIZE,,300,N;
FIN;

Output Files

Subtask 1: Positioning

DATE **/**/1997

RUN NUMBER 1000 OF 1000

CURRENT TIME .3000E+03

STATISTICAL ARRAYS CLEARED AT TIME .0000E+00

STATISTICS FOR VARIABLES BASED ON OBSERVATION

	MEAN VALUE	STAND. DEV.	COEFF. OF VARIATION	MIN VALUE	MAX VALUE	NO.OF OBS
HUM RECOVERY	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	448
HUMAN ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	39
FAILURES	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	41
SOFT RECOVERY	NO VALUES RECORDED					
SOFTWARE ERROR	NO VALUES RECORDED					
HARDWARE ERROR	NO VALUES RECORDED					
HYDRAULICS	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	2

FILE STATISTICS

FILE NUMBER	LABEL/TYPE	AVERAGE LENGTH	STANDARD DEVIATION	MAXIMUM LENGTH	CURRENT LENGTH	AVERAGE WAIT TIME
1		.000	.000	0	0	.000
2	CALENDAR	4.000	.000	4	4	133.333

REGULAR ACTIVITY STATISTICS

ACTIVITY INDEX/LABEL	AVERAGE UTILIZATION	STANDARD DEVIATION	MAX. UTIL	CURRENT UTIL	ENTITY COUNT
1	1.0000	.0000	1	1	0
2	1.0000	.0000	1	1	0
3	1.0000	.0000	1	1	0
4	1.0000	.0000	1	1	0
5	.0000	.0000	0	0	0
6	.0000	.0000	1	0	0

NOTE: The Network and Control Statements presented correspond to the output shown here for Subtask 1.

Subtask 2: Orientation

DATE **/**/1997

RUN NUMBER 1000 OF 1000

CURRENT TIME .5000E+03

STATISTICAL ARRAYS CLEARED AT TIME .3000E+03

STATISTICS FOR VARIABLES BASED ON OBSERVATION

	MEAN VALUE	STAND. DEV.	COEFF. OF VARIATION	MIN VALUE	MAX VALUE	NO.OF OBS
HUM RECOVERY	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	521
HUMAN ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	46
FAILURES	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	46
SOFT RECOVERY	NO VALUES RECORDED					
SOFTWARE ERROR	NO VALUES RECORDED					
HARDWARE ERROR	NO VALUES RECORDED					
HYDRAULICS	NO VALUES RECORDED					

FILE STATISTICS

FILE NUMBER	LABEL/TYPE	AVERAGE LENGTH	STANDARD DEVIATION	MAXIMUM LENGTH	CURRENT LENGTH	AVERAGE WAIT TIME
1		.000	.000	0	0	.000
2	CALENDAR	4.000	.000	4	4	88.889

REGULAR ACTIVITY STATISTICS

ACTIVITY INDEX/LABEL	AVERAGE UTILIZATION	STANDARD DEVIATION	MAX. UTIL	CURRENT UTIL	ENTITY COUNT
1	0.3948	.4888	1	1	0
2	0.3948	.4888	1	1	0
3	0.3948	.4888	1	1	0
4	0.3948	.4888	1	1	0
5	.0000	.0000	0	0	0
6	.0000	.0000	1	0	0

Subtask 3: Attachment

DATE **/**/1997

RUN NUMBER 1000 OF 1000

CURRENT TIME .6000E+03

STATISTICAL ARRAYS CLEARED AT TIME .5000E+03

STATISTICS FOR VARIABLES BASED ON OBSERVATION

	MEAN VALUE	STAND. DEV.	COEFF. OF VARIATION	MIN VALUE	MAX VALUE	NO.OF OBS
HUM RECOVERY	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	16
HUMAN ERROR	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	9
FAILURES	.000E+00	.000E+00	.100E+05	.000E+00	.000E+00	9
SOFT RECOVERY	NO VALUES RECORDED					
SOFTWARE ERROR	NO VALUES RECORDED					
HARDWARE ERROR	NO VALUES RECORDED					
HYDRAULICS	NO VALUES RECORDED					

FILE STATISTICS

FILE NUMBER	LABEL/TYPE	AVERAGE LENGTH	STANDARD DEVIATION	MAXIMUM LENGTH	CURRENT LENGTH	AVERAGE WAIT TIME
1		.000	.000	0	0	.000
2	CALENDAR	4.000	.000	4	4	44.444

REGULAR ACTIVITY STATISTICS

ACTIVITY INDEX/LABEL	AVERAGE UTILIZATION	STANDARD DEVIATION	MAX. UTIL	CURRENT UTIL	ENTITY COUNT
1	0.1662	.3722	1	1	0
2	0.1662	.3722	1	1	0
3	0.1662	.3722	1	1	0
4	0.1662	.3722	1	1	0
5	.0000	.0000	0	0	0
6	.0000	.0000	1	0	0

Appendix C. Plots of Rk Simulation Results (Joint d3 Locked)

The following plots were obtained by simulating various kinematic configurations over the workspace of the manipulator. Joint d3 was assumed to be locked at specific lengths, while all other joint angles were sampled from a random normal distribution. Each plot depicts Type III, Type V, and overall kinematic reliabilities. The kinematic configurations were determined by allowing each joint angle to cycle through its range of motion. The range of motion used for each joint angle is as follows:

40.0°	≤	θ1	≤	110.0°
-124.0°	≤	θ2	≤	-24.0°
74.56 in	≤	d3	≤	98.56 in
-79.9°	≤	θ4	≤	10.1°
-50.0°	≤	θ5	≤	50.0°
-80.0°	≤	θ6	≤	80.0°
80.0°	≤	θ7	≤	100.0°

In order to keep the end effector horizontal, θ4 was determined using the constraint

$$\theta_4 = -(\theta_1 + \theta_2) \quad (C.1)$$

where feasible within θ4's range. If this value fell outside of θ4's perscribed operating range, then the nearest limit was used. The configurations were determined using a number of embedded loops within the simulation program. The joint angles were varied in the following order: θ2, θ1, θ5, θ6, and finally θ7. Figure C.1 attempts to illustrate how these joint variations are represented in each of the plots.

Notice that the dips in both Type III and the overall kinematic reliability correspond to the configurations where θ5=0°. The low points in the overall kinematic reliability corresponds to configurations where θ6=0° as well.

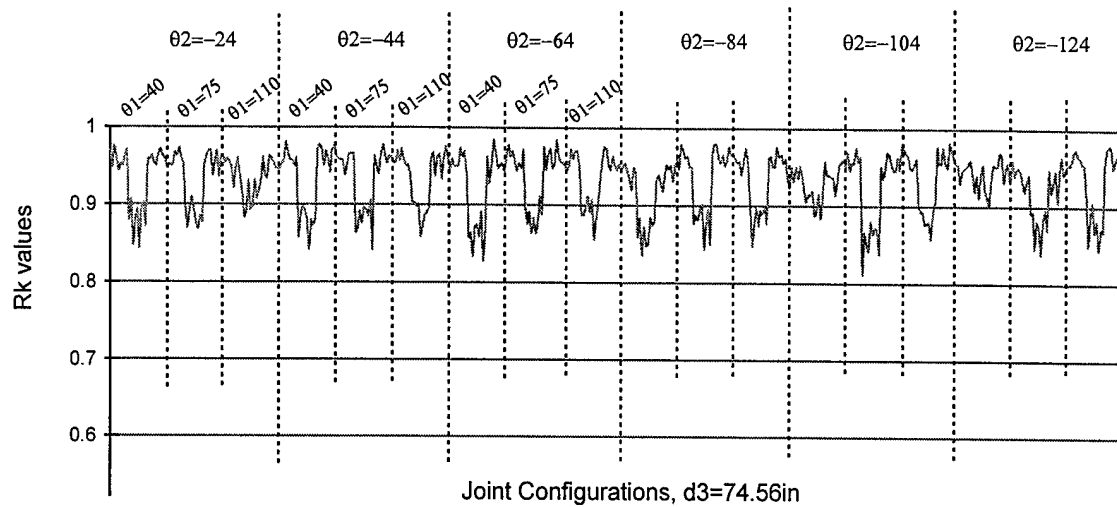


Figure C.1 Joint Variations within Kinematic Configurations

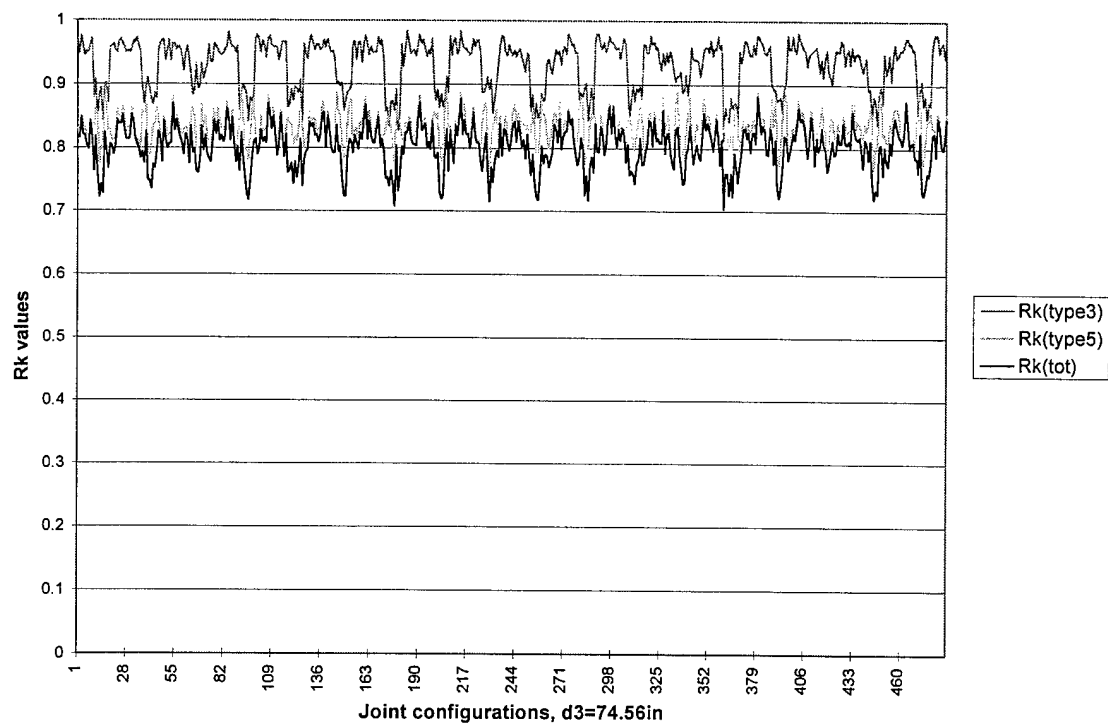


Figure C.2 R_k Values for Joint Configurations ($d_3=74.56\text{in}$)

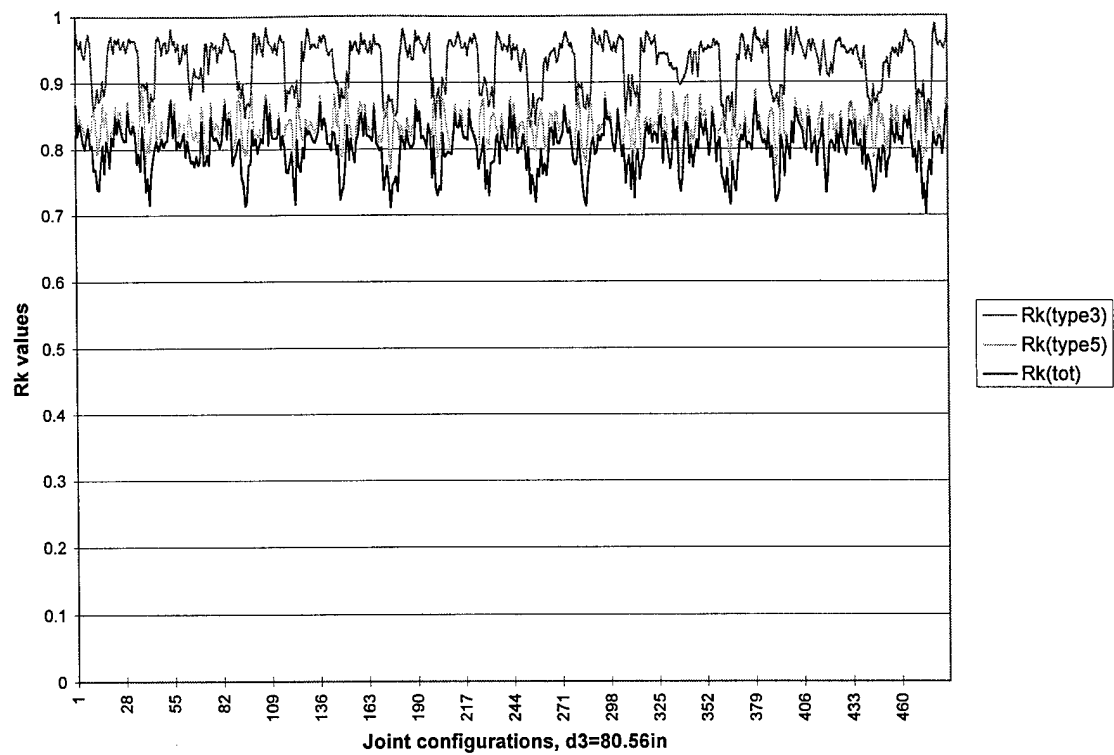


Figure C.3 R_k Values for Joint Configurations ($d_3=80.56\text{in}$)

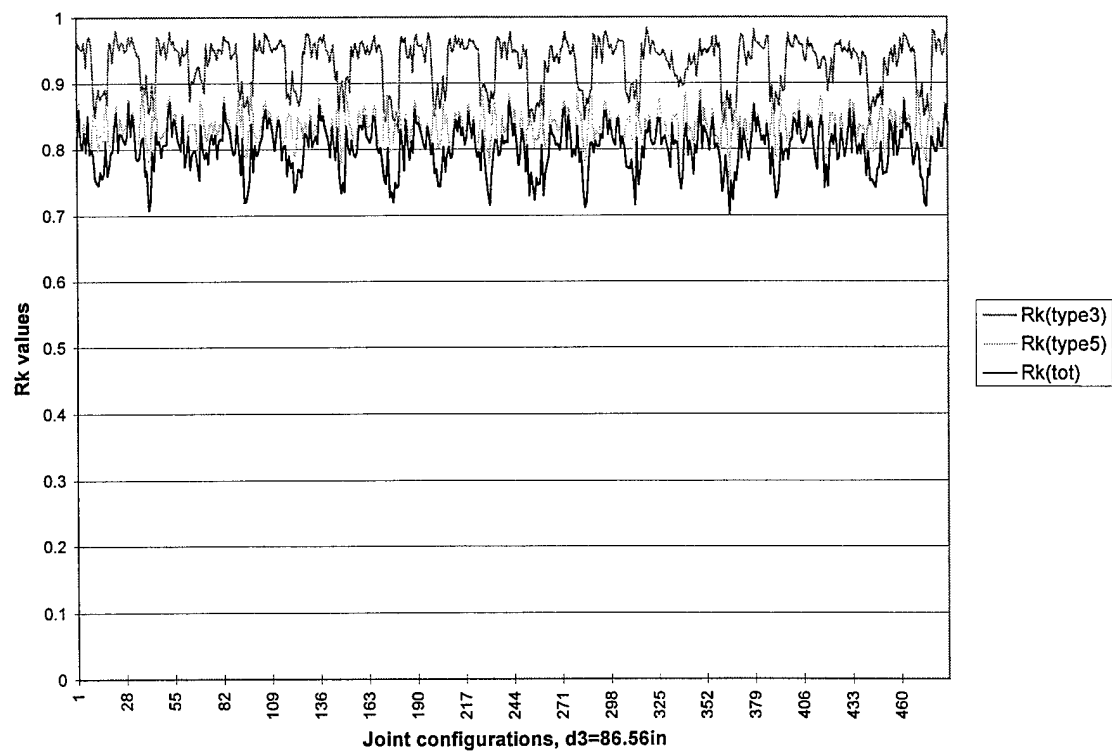


Figure C.4 R_k Values for Joint Configurations ($d_3=86.56\text{in}$)

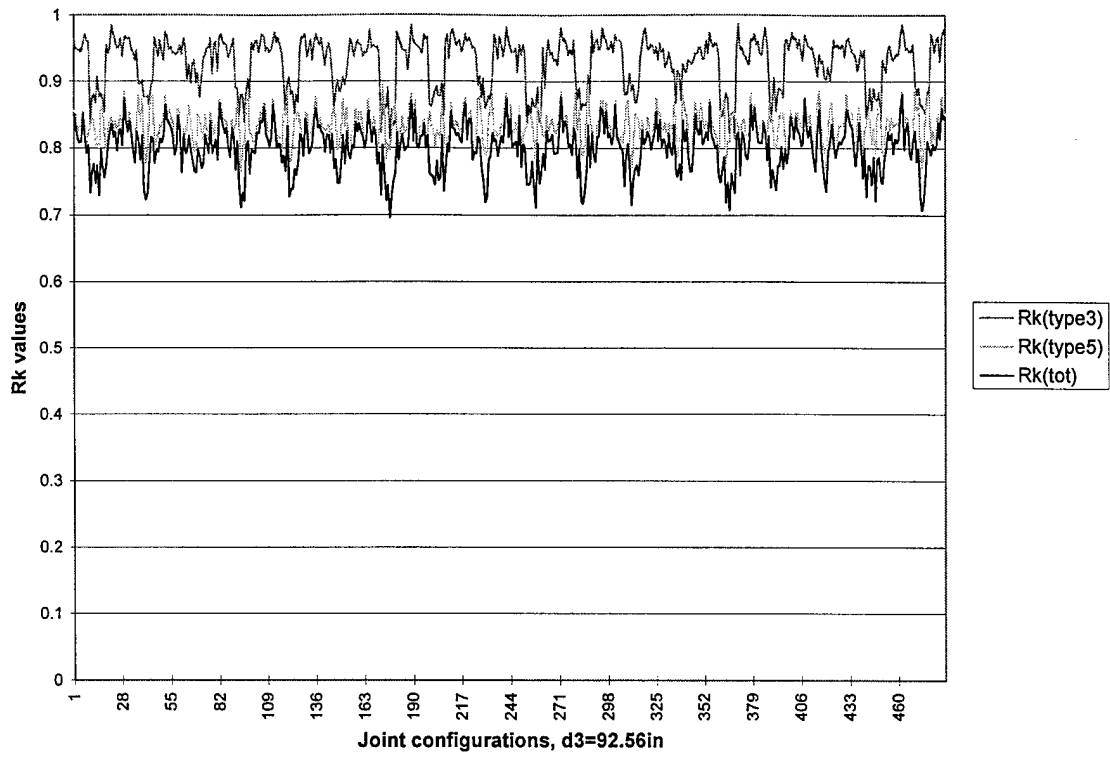


Figure C.5 R_k Values for Joint Configurations ($d3=92.56\text{in}$)

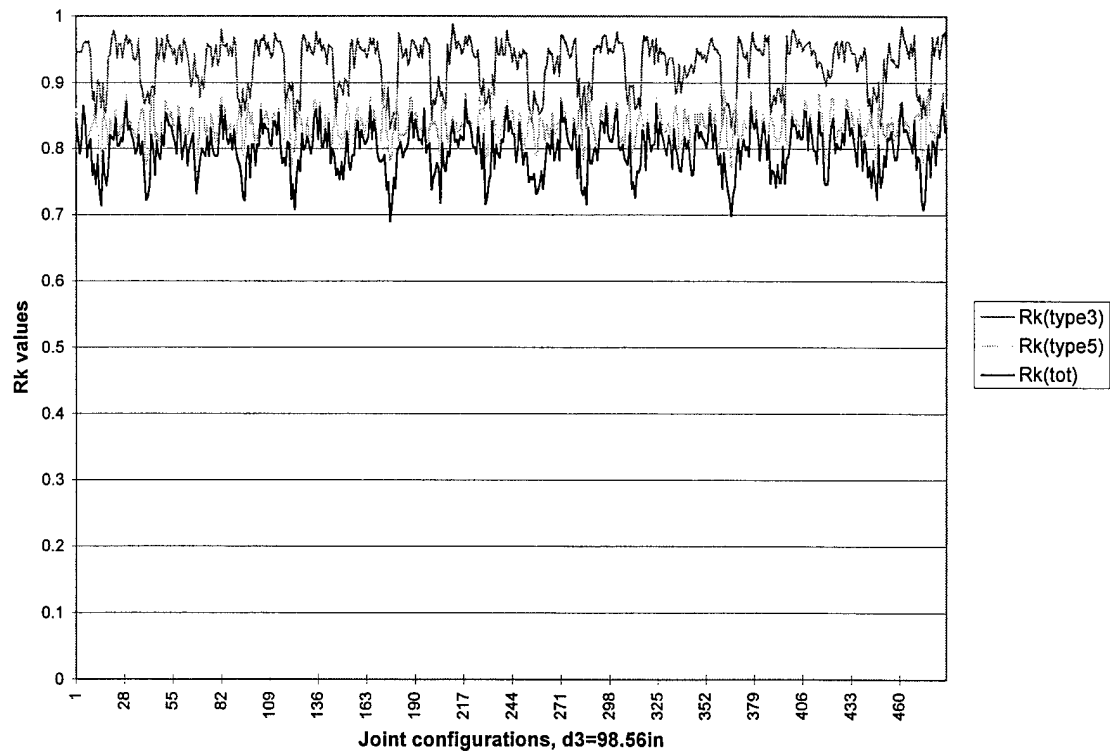


Figure C.6 R_k Values for Joint Configurations ($d3=98.56\text{in}$)

Appendix D. Plots of R_k Simulation Results (θ_1 Locked)

These plots are similar to those shown in Appendix C. Now θ_1 is assumed to be set exactly at certain angles while all other joint variables are sampled from a random normal distribution. The embedded loops within the simulation are now performed in the following order: θ_1 , θ_2 , d_3 , θ_5 , θ_6 , and finally θ_7 . Figure D.1 attempts to illustrate the pattern of angle variations contained within each plot.

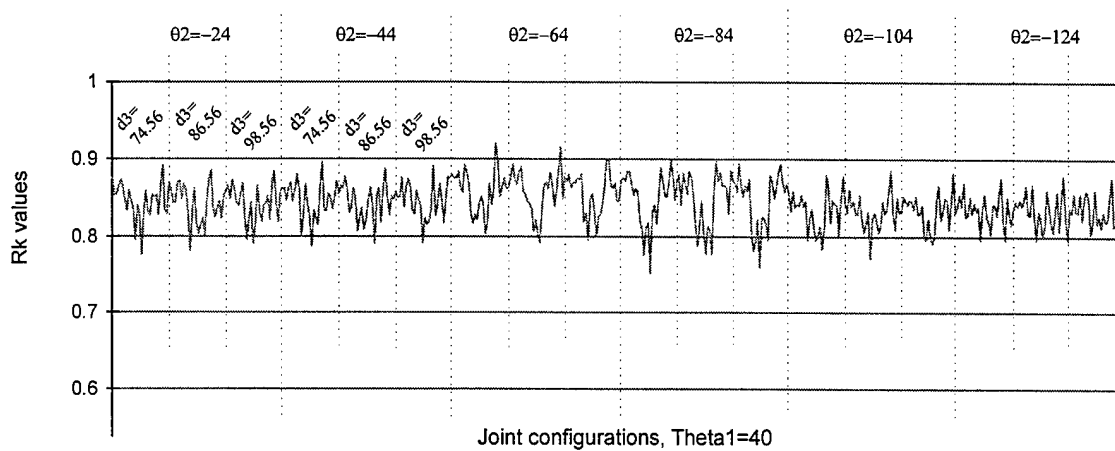


Figure D.1 Joint Variations within Kinematic Configurations

One of the trends seen within the following plots are lower estimates of kinematic reliability when θ_5 is at or near 0° . There is a significant degree of variability for all three types of kinematic reliability. The shape of $R_k(\text{tot})$ corresponds fairly well with that of the $R_k(\text{III})$ estimates.

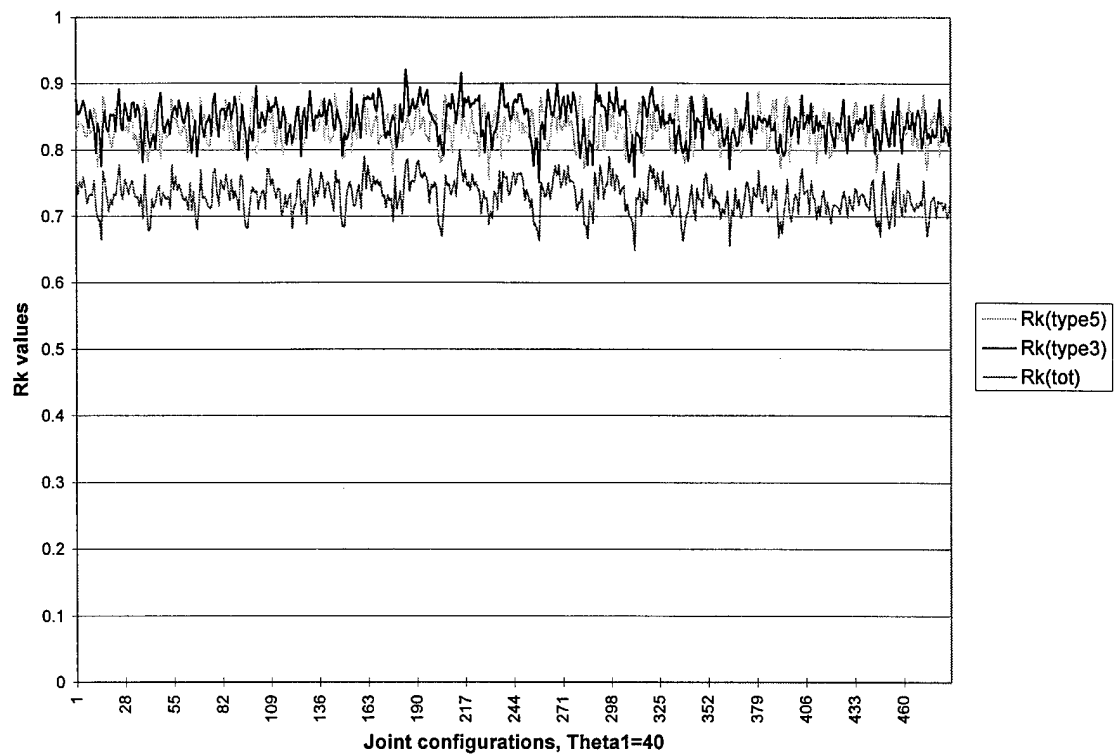


Figure D.2 R_k Values for Joint Configurations ($\theta_1 = 40^\circ$)

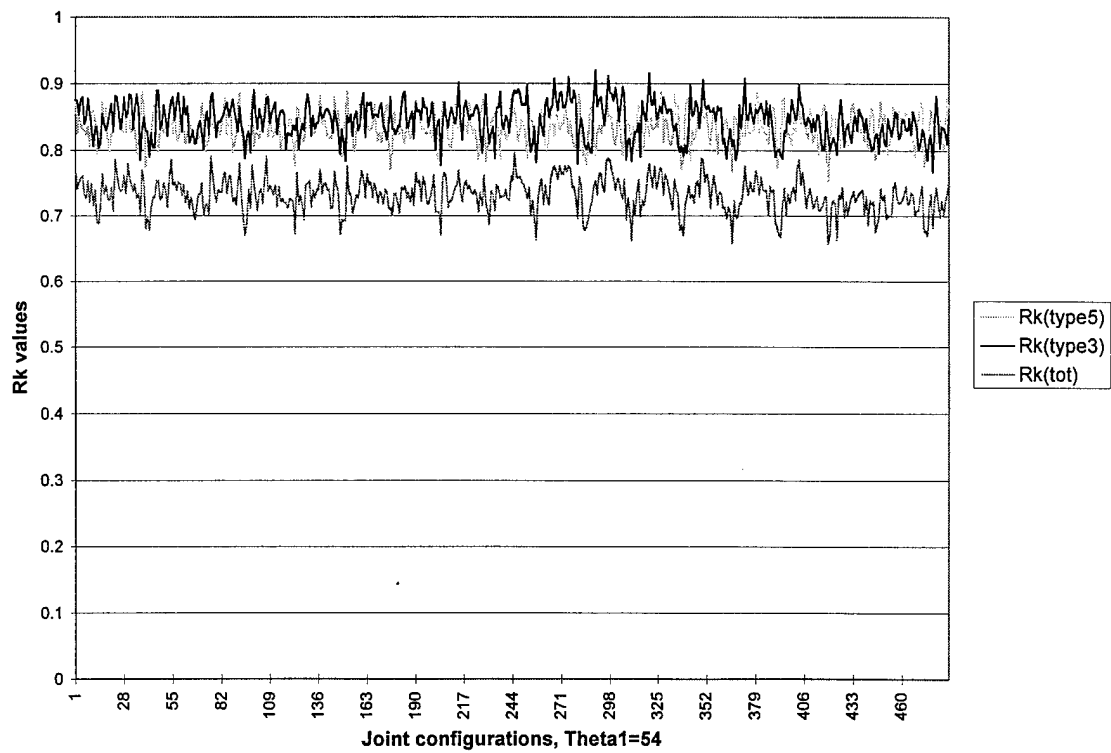


Figure D.3 R_k Values for Joint Configurations ($\theta_1 = 54^\circ$)

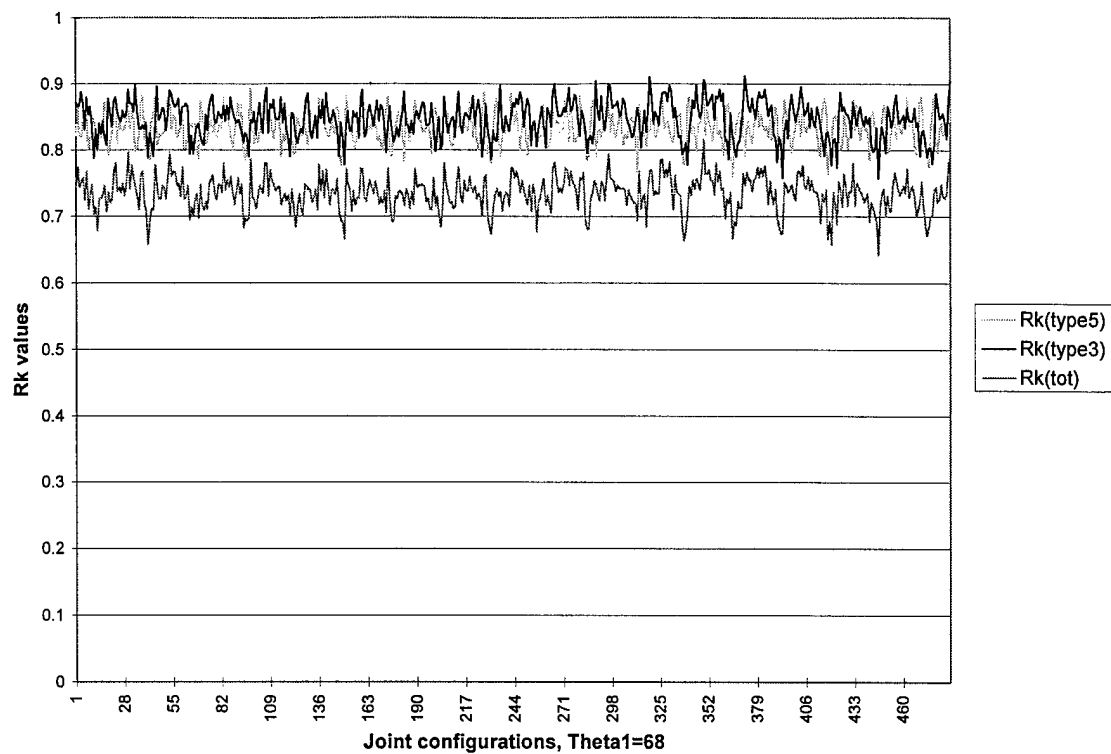


Figure D.4 R_k Values for Joint Configurations ($\theta_1=68^\circ$)

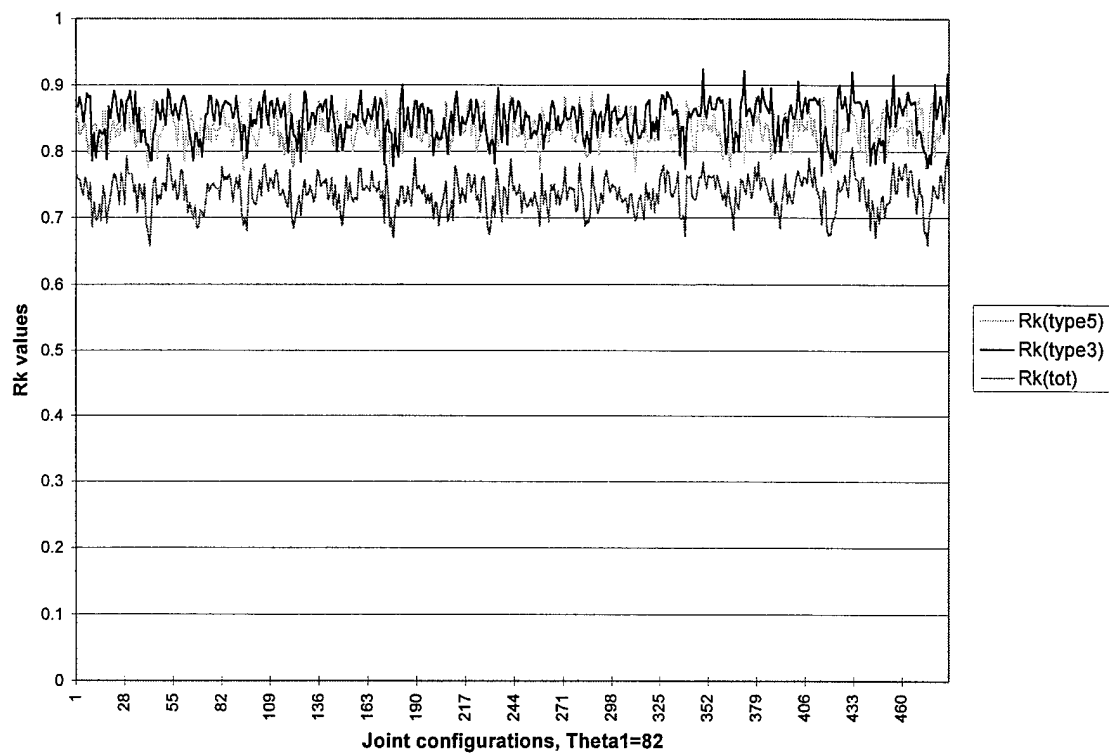


Figure D.5 R_k Values for Joint Configurations ($\theta_1=82^\circ$)

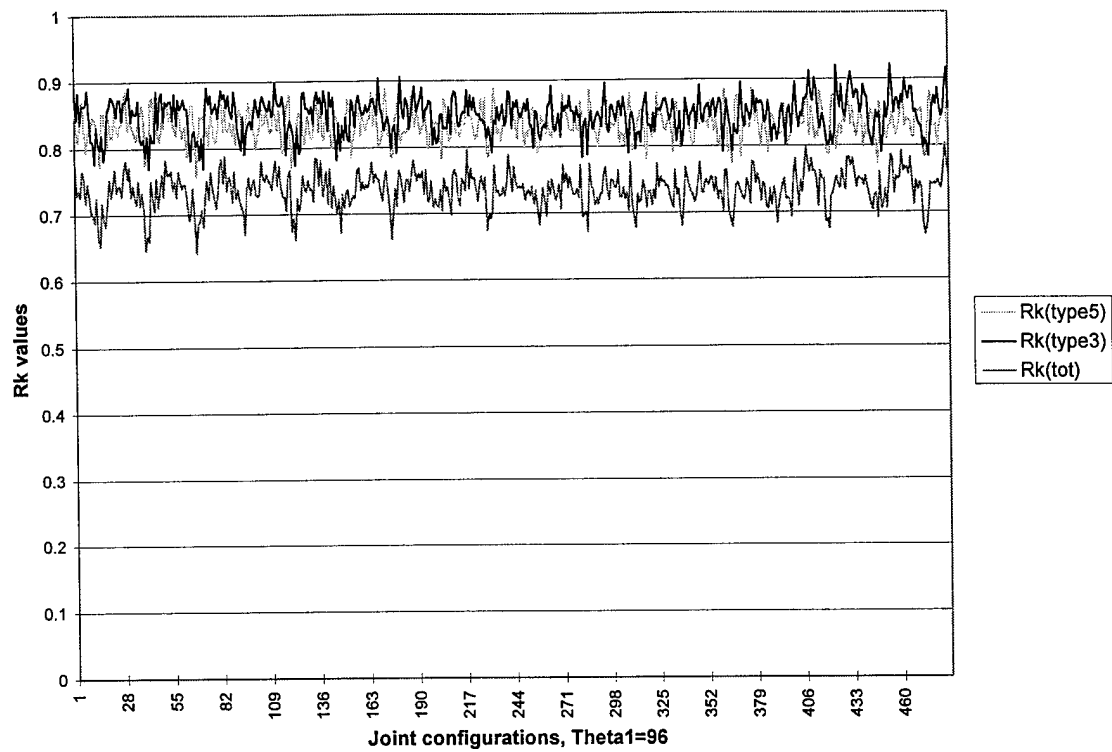


Figure D.6 R_k Values for Joint Configurations ($\theta_1=96^\circ$)

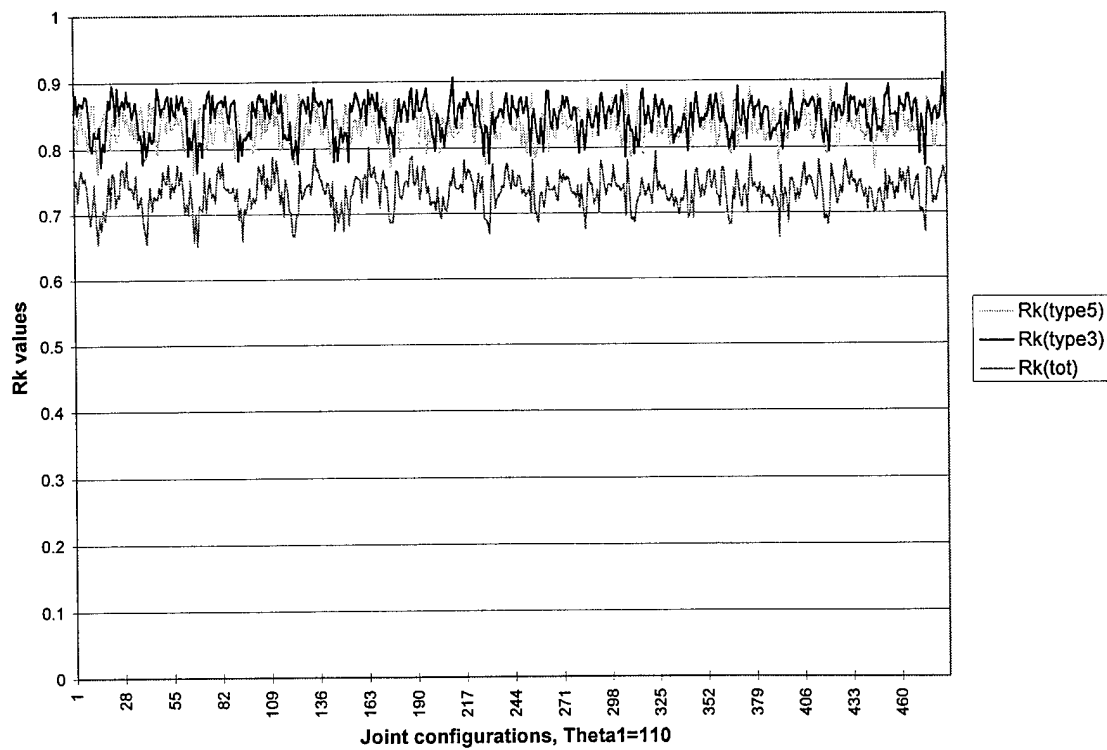


Figure D.7 R_k Values for Joint Configurations ($\theta_1=110^\circ$)

Appendix E. FORTRAN Program for Rk Calculations

```
PROGRAM RKCALC
INTEGER j,n,k,l,m,t,u,v
REAL rkt3,rkt5,rktot,totsuc,totboth,totrota
REAL sig1,sig2,del3,sig4,sig5,sig6,sig7
REAL conv,xcount,ycount,zcount
real AE,A5,D5,L1,L4,O4,AEP
real theta1,theta2,d3,theta4,theta5,theta6,theta7
real theta2P,theta4P,xd,yd,zd
real t1norm,t2Pnorm,d3norm,t4Pnorm,t5norm,t6norm
real t7norm,xpos,ypos,zpos
real c56,c7P,c5,c12P,c1,c2P,c12P4P,s12P4P,c8
real s56,s7P,s5,s12P,s1,s2P,s8
real c56n,c7Pn,c5n,c12Pn,c1n,c2Pn,c12P4Pn
real s12P4Pn,s56n,s7Pn,s5n,s12Pn,s1n,s2Pn
real r11,r21,r31,r12,r22,r32,r13,r23,r33
real pitch,roll,yaw,pitchn,rolln,yawn
real picount,rocount,yacount,type3,type5

** Conversion from degrees to rads
conv = 0.0174532925

** Number of runs within simulation
n = 500

** All units are in m
AE = 0.10795
A5 = 0.6096
D5 = 0.20955
L1 = 0.5334
L4 = 0.127
O4 = 0.13335
AEP= 0.0

** All angles converted to rads
sig1 = .00549316*conv
sig2 = .00549316*conv
del3 = .001
sig4 = .0878906*conv
sig5 = .0878906*conv
sig6 = .0878906*conv
sig7 = .0878906*conv

** Begin loops for joint variables
theta1=40*conv

DO 20 k=1,6
theta2=-24.0*conv

DO 30 l=1,6
d3=74.56*.0254

** Constraint to keep end effector horizontal
theta4=-(theta1+theta2)
IF (theta4 .GT. (10.1*conv)) theta4=10.1*conv
IF (theta4 .LT. (-79.9*conv)) theta4=-79.9*conv
```

```

** Convert to th2P and th4P used in equations
    theta2P=theta2-1.5708
    theta4P=theta4+1.5708

DO 40 m=1,3
    theta5=50.0*conv

DO 50 t=1,3
    theta6=80.0*conv

DO 60 u=1,3
    theta7=100.0*conv

DO 70 v=1,3

***** TRIG CALLS *****
    c56 = cos(theta5 + theta6)
    s56 = sin(theta5 + theta6)
    c7P = cos(theta7)
    s7P = sin(theta7)
    c5 = cos(theta5)
    s5 = sin(theta5)
    c12P= cos(theta1 + theta2P)
    s12P= sin(theta1 + theta2P)
    c1 = cos(theta1)
    s1 = sin(theta1)
    c2P = cos(theta2P)
    s2P = sin(theta2P)
    c12P4P = cos(theta1 + theta2P + theta4P)
    s12P4P = sin(theta1 + theta2P + theta4P)
    c8 = 1.0
    s8 = 0.0

***** Desired Cartesian x, y, z Calc *****
* x of end effector wrpt frame {0}
    xd = AEP*c12P4P*s56*s8 + (AEP*c8+AE)*(c12P4P*c56*c7P
    & - s12P4P*s7P) + A5*c12P4P*c5 - D5*s12P4P + L4*c12P4P
    & - d3*s12P + L1*c1 + O4*c12P

* y of end effector wrpt frame {0}
    yd = AEP*s12P4P*s56*s8 + (AEP*c8 + AE)*(s12P4P*c56*c7P
    & + c12P4P*s7P) + A5*s12P4P*c5 + D5*c12P4P + L4*s12P4P
    & + d3*c12P + L1*s1 + O4*s12P

* z of end effector wrpt frame {0}
    zd = AEP*(-s56*c7P*c8 + c56*s8) -AE*s56*c7P - A5*s5

***** Rotation Matrix Calc *****
    r11 = c12P4P*c56*c7P*c8 - s12P4P*s7P*c8 + c12P4P*s56*s8
    r21 = s12P4P*c56*c7P*c8 + c12P4P*s7P*c8 + s12P4P*s56*s8
    r31 = -s56*c7P*c8 + c56*s8

    r12 = -c12P4P*c56*s7P -s12P4P*c7P
    r22 = -s12P4P*c56*s7P +c12P4P*c7P
    r32 = s56*s7P

    r13 = -c12P4P*c56*c7P*s8 + s12P4P*s7P*s8 + c12P4P*s56*c8
    r23 = -s12P4P*c56*c7P*s8 - c12P4P*s7P*s8 + s12P4P*s56*c8

```

```

r33 = s56*c7P*s8 + c56*c8

*** Pitch Roll and Yaw Calculations (degs) ***

pitch = (ATAN2(r21,r11))/conv
roll = (ATAN2(-r31,(r11*cos(pitch*conv)+
& r21*sin(pitch*conv)))/conv
yaw = (ATAN2((r13*sin(pitch*conv)-r23*cos(pitch*conv)),
& (r22*cos(pitch*conv)-r12*sin(pitch*conv)))/conv

**** BEGIN SIMULATION FOR JOINT VARIABLES ****

** Initialize total success counters to zero
totsuc = 0.0
totrota = 0.0
totboth = 0.0

do 10 j = 1,n

    xcount = 0.0
    ycount = 0.0
    zcount = 0.0
    picount = 0.0
    rocount = 0.0
    yacount = 0.0
    type3 = 0.0
    type5 = 0.0

** Theta1 assumed to be exactly fixed
tlnorm = theta1

** Joint variables sampled from random normal
CALL RNORM(theta2P,sig2,t2Pnorm)
CALL RNORM(d3,d3,d3norm)
CALL RNORM(theta4P,sig4,t4Pnorm)
CALL RNORM(theta5,sig5,t5norm)
CALL RNORM(theta6,sig6,t6norm)
CALL RNORM(theta7,sig7,t7norm)

***** TRIG CALLS *****
c56n = cos(t5norm + t6norm)
s56n = sin(t5norm + t6norm)
c7Pn = cos(t7norm)
s7Pn = sin(t7norm)
c5n = cos(t5norm)
s5n = sin(t5norm)
c12Pn= cos(tlnorm + t2Pnorm)
s12Pn= sin(tlnorm + t2Pnorm)
c1n = cos(tlnorm)
s1n = sin(tlnorm)
c2Pn = cos(t2Pnorm)
s2Pn = sin(t2Pnorm)
c12P4Pn = cos(tlnorm + t2Pnorm + t4Pnorm)
s12P4Pn = sin(tlnorm + t2Pnorm + t4Pnorm)

***** Cartesian x, y, z Calc *****
* x of end effector wrpt frame {0}
xpos = AEP*c12P4Pn*s56n*s8 + (AEP*c8+AE)*(c12P4Pn*c56n*c7Pn
& - s12P4Pn*s7Pn) + A5*c12P4Pn*c5n - D5*s12P4Pn + L4*c12P4Pn

```



```

& - d3norm*s12Pn + L1*c1n + O4*c12Pn

* y of end effector wrpt frame {0}
ypos = AEP*s12P4Pn*s56n*s8 + (AEP*c8 + AE)*(s12P4Pn*c56n*c7Pn
& + c12P4Pn*s7Pn) + A5*s12P4Pn*c5n + D5*c12P4Pn + L4*s12P4Pn
& + d3norm*c12Pn + L1*s1n + O4*s12Pn

* z of end effector wrpt frame {0}
zpos = AEP*(-s56n*c7Pn*c8 + c56n*s8) -AE*s56n*c7Pn - A5*s5n

rn11 = c12P4Pn*c56n*c7Pn*c8 - s12P4Pn*s7Pn*c8
& + c12P4Pn*s56n*s8
rn21 = s12P4Pn*c56n*c7Pn*c8 + c12P4Pn*s7Pn*c8
& + s12P4Pn*s56n*s8
rn31 = -s56n*c7Pn*c8 + c56n*s8

rn12 = -c12P4Pn*c56n*s7Pn -s12P4Pn*c7Pn
rn22 = -s12P4Pn*c56n*s7Pn +c12P4Pn*c7Pn
rn32 = s56n*s7Pn

rn13 = -c12P4Pn*c56n*c7Pn*s8 + s12P4Pn*s7Pn*s8
& + c12P4Pn*s56n*c8
rn23 = -s12P4Pn*c56n*c7Pn*s8 - c12P4Pn*s7Pn*s8
& + s12P4Pn*s56n*c8
rn33 = s56n*c7Pn*s8 + c56n*c8

pitchn = (ATAN2(rn21,rn11))/conv
rolln = (ATAN2(-rn31,(rn11*cos(pitchn*conv)+
& rn21*sin(pitchn*conv))))/conv
yaw = (ATAN2((rn13*sin(pitchn*conv)-rn23*cos(pitchn*conv)),
& (rn22*cos(pitchn*conv)-rn12*sin(pitchn*conv))))/conv

*** Check bounds for position ***
IF (xpos .GT. (xd-.002) .AND. xpos .LT. (xd+.002)) THEN
  xcount = 1.0
END IF

IF (ypos .GT. (yd-.002) .AND. ypos .LT. (yd+.002)) THEN
  ycount = 1.0
END IF

IF (zpos .GT. (zd-.002) .AND. zpos .LT. (zd+.002)) THEN
  zcount = 1.0
END IF

IF (xcount .EQ. 1.0 .AND. ycount .EQ. 1.0 .AND. zcount .EQ. 1.0)
& THEN
  totsuc=totsuc+1.0
  type3=1.0
END IF

*** Check bounds for orientation ***
IF(pitchn .GT. (pitch-.2) .AND. pitchn .LT. (pitch+.2)) THEN
  picount=1.0
END IF

IF (rolln .GT. (roll-.2) .AND. rolln .LT. (roll+.2)) THEN
  rocount=1.0
END IF

```

```

      IF (yawn .GT. (yaw-.2) .AND. yawn .LT. (yaw+.2)) THEN
        yacount=1.0
      END IF

      IF (picount .EQ. 1.0 .AND. rocount .EQ. 1.0 .AND. yacount .EQ.
& 1.0) THEN
        totrota=totrota+1.0
        type5=1.0
      END IF

      IF (type3 .EQ. 1.0 .AND. type5 .EQ. 1.0) totboth=totboth+1.0

10  continue

      rkt3 = totsuc/n
      rkt5 = totrota/n
      rktot = totboth/n

      print 95, (theta1/conv), (theta2/conv), (d3/.0254), (theta4/conv),
& (theta5/conv), (theta6/conv), (theta7/conv), xd, yd, zd, rkt3, rkt5,
& rktot

95  format (f5.1, f7.1, f6.2, f7.2, 3f6.1, f6.3, 2f7.3, 2f6.3, f5.3)

      theta7=theta7-(10*conv)
70  continue

      theta6=theta6-(80*conv)
60  continue

      theta5=theta5-(50*conv)
50  continue

      d3=d3+(12.0*.0254)
40  continue

      theta2=theta2-(20*conv)
30  continue

      theta1=theta1+(14*conv)
20  continue

      END

```

```
SUBROUTINE RNORM(mean,sigma,tnorm)
INTEGER iset
real r1,r2,a,b,w,RAND,mean,sigma,tnorm

50  IF (iset .EQ. 0) THEN
      r1=RAND(637378)
      r2=RAND(737828)
      a=(2.0*r1)-1.0
      b=(2.0*r2)-1.0
      w=(a*a)+(b*b)
      IF (w .GT. 1.0) GOTO 50

      tnorm = mean + (sigma*(a*SQRT(-2.0*ALOG(w)/w)))
      iset = 1

      ELSE

      tnorm = mean + (sigma*(b*SQRT(-2.0*ALOG(w)/w)))
      iset = 0

      END IF

END

FUNCTION RAND(K)

INTEGER K,M,CONST1
REAL RAND,CONST2
PARAMETER (CONST1=2147483645, CONST2=.4656613E-9)
SAVE
DATA M/0/

IF (M .EQ. 0) M=K
M = M*65539
IF (M .LT. 0) M = (M+1) + CONST1
RAND = M *CONST2
END
```

Appendix F. Schematic of Joint Variables for ATD System

Please note that the following illustrations are not to scale. They are intended to provide insight regarding how the joint angles/lengths affect the position and orientation of the end-effector.

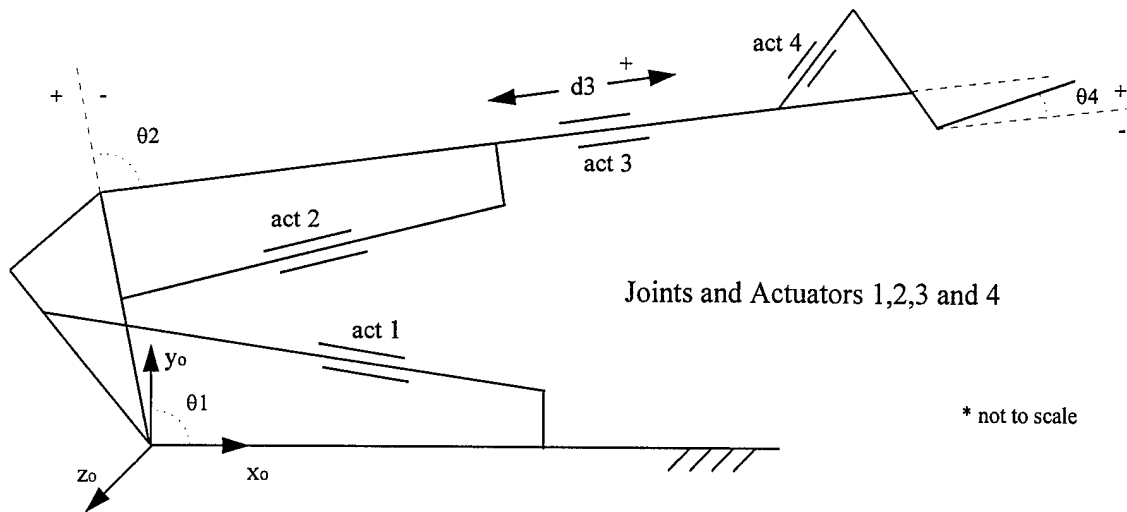


Figure F.1 Schematic for First Four Joints

Joint Constraints				
40.0°	≤	θ1	≤	130.0°
-134.0°	≤	θ2	≤	-24.0°
74.56 in	≤	d3	≤	98.56 in
-79.9°	≤	θ4	≤	10.1°

NOTE: $\theta_2' = \theta_2 - 90^\circ$
 $\theta_4' = \theta_4 + 90^\circ$

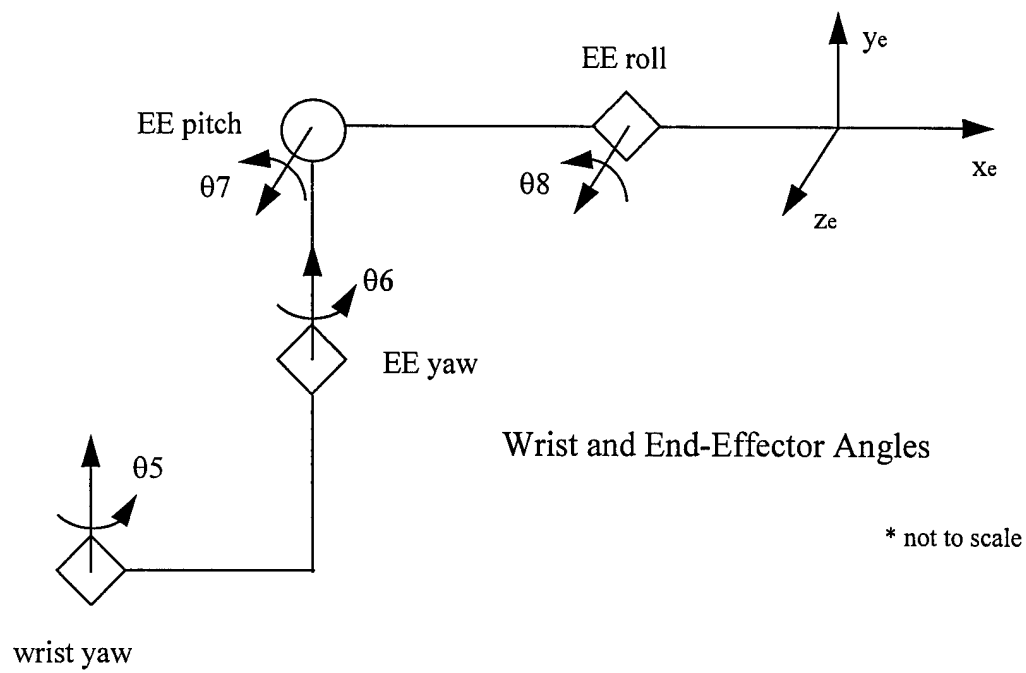


Figure F.2 Schematic for Last Four Joints

<u>Joint Constraints</u>				
-50.0°	≤	θ5	≤	50.0°
-200.0°	≤	θ6	≤	200.0°
80.0°	≤	θ7	≤	100.0°
θ8 is a continuous roll				

Appendix G. Pitch, Roll, and Yaw Angle Formulation

The formulation of the pitch, roll, and yaw angles (denoted as θ , ϕ , and ψ respectively) is presented here. This formulation follows a method presented by Spong in [32]. These angles represent the orientation of the end-effector with respect to the base coordinate frame. The order of rotation is as follows: first a pitch about the z_0 axis through an angle θ , then a roll about the x_0 axis through an angle ϕ , and finally a yaw about the y_0 axis through an angle ψ (see Figure G.1). Since the successive rotations are relative to the fixed base frame, a transformation matrix can be found using

$$R_0^1 = R_{y,\psi} R_{x,\phi} R_{z,\theta} \quad (G.1)$$

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} c\psi & 0 & s\psi \\ 0 & 1 & 0 \\ -s\psi & 0 & c\psi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The calculations for the R_0^1 matrix (**n**, **s**, and **a** vectors) were provided by ORNL and can be found within the FORTRAN simulation developed for the R_k calculations (see Appendix E). The pitch, roll, and yaw angles can then be found as follows

$$R_{z,\theta}^T R_0^1 = R_{y,\psi} R_{x,\phi} \quad (G.2)$$

$$\begin{bmatrix} (r_{11}c\theta + r_{21}s\theta) & (r_{12}c\theta + r_{22}s\theta) & (r_{13}c\theta + r_{23}s\theta) \\ (-r_{11}s\theta + r_{21}c\theta) & (-r_{12}s\theta + r_{22}c\theta) & (-r_{13}s\theta + r_{23}c\theta) \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} c\psi & s\psi s\phi & s\psi c\phi \\ 0 & c\phi & -s\phi \\ -s\psi & c\psi s\phi & c\psi c\phi \end{bmatrix}$$

note: $R_{z,\theta}$ is an orthogonal transform (i.e. $R_{z,\theta}^{-1} = R_{z,\theta}^T$)
 $c = \text{cosine}$, $s = \text{sine}$

$$\begin{aligned}
 -r_{11}s\theta + r_{21}c\theta &= 0 \Rightarrow \frac{r_{21}}{r_{11}} = \frac{s\theta}{c\theta} \\
 \theta &= \arctan\left(\frac{r_{21}}{r_{11}}\right) = \text{ATAN2}(r_{21}, r_{11})
 \end{aligned}
 \tag{G.3}$$

$$\begin{aligned}
 \left. \begin{aligned} c\psi &= r_{11}c\theta + r_{21}s\theta \\ s\psi &= -r_{31} \end{aligned} \right\} \Rightarrow \frac{s\psi}{c\psi} = \frac{-r_{31}}{r_{11}c\theta + r_{21}s\theta} \\
 \psi &= \text{ATAN2}(-r_{31}, r_{11}c\theta + r_{21}s\theta)
 \end{aligned}
 \tag{G.4}$$

$$\begin{aligned}
 \left. \begin{aligned} c\phi &= -r_{12}s\theta + r_{22}c\theta \\ s\phi &= r_{13}s\theta - r_{23}c\theta \end{aligned} \right\} \Rightarrow \frac{s\phi}{c\phi} = \frac{r_{13}s\theta - r_{23}c\theta}{-r_{12}s\theta + r_{22}c\theta} \\
 \phi &= \text{ATAN2}(r_{13}s\theta - r_{23}c\theta, -r_{12}s\theta + r_{22}c\theta)
 \end{aligned}
 \tag{G.5}$$

From Equation G.3 it is apparent that if both r_{11} and r_{21} become zero, the angle θ becomes undefined. However, for the range of joint variables used in the R_k calculations, Equation G.3 appears to work fine.

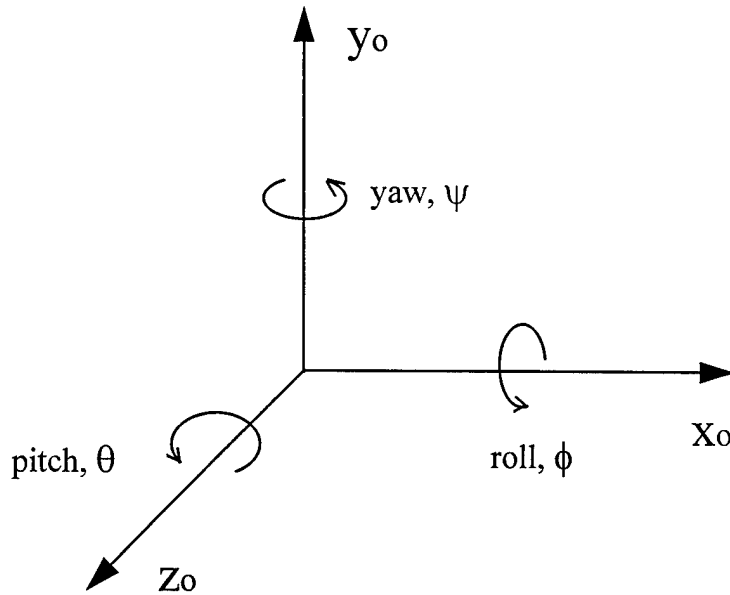


Figure G.1 Pitch, Roll, and Yaw Angles

Appendix H. Trajectories for ATD System

This appendix gives a more complete description of the three arbitrary trajectories used in the task analysis of the ATD system. Recall that these trajectories are depicted in Figure 4.3. The length variable, d_3 , is assumed to be exact for all three trajectories. The other joint variables shown were all sampled from a random normal distribution to calculate the R_k values. All values given in the tables are the mean or desired values. Note that Type III kinematic reliability is being used for the positioning subtask, while total kinematic reliability (Type III and Type V) is used for the orientation.

Table H.1 Trajectory 1 Positioning

θ_1	θ_2	d_3	θ_4	θ_5	θ_6	θ_7	x	y	z	Rk(III)
110	-64	80.56	-46	0	0	90	2.0715	2.198	0	0.9
100	-44	80.56	-56	0	0	90	1.8988	2.4646	0	0.904
80	-24	80.56	-56	0	0	90	2.084	2.4646	0	0.868
70	-24	80.56	-46	0	0	90	2.4364	2.198	0	0.908
80	-44	80.56	-36	0	0	90	2.563	1.9376	0	0.882
90	-64	80.56	-26	0	0	90	2.6342	1.628	0	0.908
60	-44	80.56	-16	0	0	90	3.007	1.2153	0	0.882
70	-64	80.56	-6	0	0	90	2.968	0.9	0	0.862
60	-64	80.56	4	0	0	90	3.0352	0.5037	0	0.89
40	-44	80.56	4	0	0	90	3.1771	0.3846	0	0.888
50	-44	80.56	-6	0	0	90	3.1284	0.8074	0	0.85

Table H.2 Trajectory 1 Orientation

θ_1	θ_2	d_3	θ_4	θ_5	θ_6	θ_7	x	y	z	Rk(III)	Rk(V)	Rk(tot)
50	-44	80.56	-6	-10	0	90	3.119	0.807	0.106	0.908	0.816	0.764
50	-44	80.56	-6	-20	0	90	3.092	0.807	0.208	0.898	0.79	0.74
50	-44	80.56	-6	-20	20	90	3.092	0.807	0.208	0.91	0.776	0.744
50	-44	80.56	-6	-20	20	100	3.073	0.806	0.208	0.926	0.792	0.748
50	-44	80.56	-6	-20	40	100	3.074	0.806	0.215	0.926	0.814	0.77

Table H.3 Trajectory 2 Positioning

$\theta 1$	$\theta 2$	d3	$\theta 4$	$\theta 5$	$\theta 6$	$\theta 7$	x	y	z	Rk(III)
110	-64	74.56	-46	0	0	90	1.9657	2.0884	0	0.906
80	-24	74.56	-56	0	0	90	1.9988	2.3383	0	0.866
70	-24	74.56	-46	0	0	90	2.3305	2.0884	0	0.884
80	-44	74.56	-36	0	0	90	2.4397	1.8481	0	0.88
80	-64	74.56	-16	0	0	90	2.6864	1.2366	0	0.88
80	-84	74.56	4	0	0	90	2.7091	0.5777	0	0.888
40	-24	74.56	-16	0	0	90	3.0024	1.0542	0	0.89

Table H.4 Trajectory 2 Orientation

$\theta 1$	$\theta 2$	d3	$\theta 4$	$\theta 5$	$\theta 6$	$\theta 7$	x	y	z	Rk(III)	Rk(V)	Rk(tot)
40	-24	74.56	-16	-10	0	90	2.993	1.054	0.106	0.906	0.816	0.762
40	-24	74.56	-16	-20	0	90	2.966	1.054	0.208	0.904	0.79	0.744
40	-24	74.56	-16	-20	20	90	2.966	1.054	0.208	0.912	0.776	0.746
40	-24	74.56	-16	-20	20	100	2.947	1.053	0.208	0.924	0.792	0.748
40	-24	74.56	-16	-20	40	100	2.948	1.053	0.215	0.934	0.814	0.776

Table H.5 Trajectory 3 Positioning

$\theta 1$	$\theta 2$	d3	$\theta 4$	$\theta 5$	$\theta 6$	$\theta 7$	x	y	z	Rk(III)
110	-44	86.56	-66	0	0	90	1.5703	2.773	0	0.892
100	-44	86.56	-56	0	0	90	1.984	2.591	0	0.866
100	-64	86.56	-36	0	0	90	2.5011	2.0272	0	0.874
110	-104	86.56	-6	0	0	90	2.7547	0.9159	0	0.878
80	-64	86.56	-16	0	0	90	2.9794	1.3206	0	0.884

Table H.6 Trajectory 3 Orientation

$\theta 1$	$\theta 2$	d3	$\theta 4$	$\theta 5$	$\theta 6$	$\theta 7$	x	y	z	Rk(III)	Rk(V)	Rk(tot)
80	-64	86.56	-16	-10	0	90	2.97	1.321	0.106	0.908	0.816	0.764
80	-64	86.56	-16	-20	0	90	2.943	1.321	0.208	0.9	0.79	0.74
80	-64	86.56	-16	-20	20	90	2.943	1.321	0.208	0.908	0.776	0.742
80	-64	86.56	-16	-20	20	100	2.924	1.319	0.208	0.928	0.792	0.75
80	-64	86.56	-16	-20	40	100	2.925	1.319	0.215	0.928	0.814	0.772

Notes: The minimum R_k values used in the reliability index calculations are shown in boldface lettering.

Angles are in degrees, d3 given in inches, and the x,y, and z coordinates are in meters as measured from the base reference frame.

Bibliography

- [1] Abbas, B.S. and Kuo, W., "Stochastic Effectiveness Models for Human-Machine Systems," *IEEE Transactions on System, Man, and Cybernetics*, vol. 20, pp. 826-834, 1990.
- [2] Bhatti, P., *Probabilistic Modeling and Optimal Design of Robotic Manipulators*, PhD dissertation, Purdue University, West Lafayette, IN, 1989 (ON9018787).
- [3] Bhatti, P. and Rao, S., "Reliability Analysis of Robot Manipulators," *Journal of Mechanisms, Transmissions, and Automation in Design*, vol.110, pp.175-181, June 1988.
- [4] Cacciabue, P.C., Carpignano, A., and Vivalda, C., "A Dynamic Reliability Technique for Error Assessment in Man-Machine Systems," *International Journal of Man-Machine Studies*, vol. 38, pp. 403-428, 1993.
- [5] Chesser, J. and Jansen, J.F., "Control Strategies for the Next Generation Munition Handler," Unpublished Report, Oak Ridge National Laboratory, Oak Ridge, TN, 1996.
- [6] Dhillon, B. and Yang, N., "Human Reliability: A Literature Survey and Review," *Microelectronics and Reliability*, vol.34, no.5, pp. 803-810, 1994.
- [7] Dhillon, B., *Human Reliability with Human Factors*, Pergamon Press, New York, 1986.
- [8] Dhillon, B., *Robot Reliability and Safety*, Springer-Verlag, New York, 1991.
- [9] Dougherty, E.M. and Fragola, J.R., *Human Reliability Analysis: A Systems Engineering Approach with Nuclear Power Plant Applications*, John Wiley and Sons, New York, 1988.
- [10] Green, D.E., Barr, R.E., Fulcher, C., Hwang, L., and Rao, S., "A Stochastic Sequential Model for Man-Machine Tracking Systems," *IEEE Transactions on System, Man, and Cybernetics*, vol. 18, no. 2, pp. 316-326, March/April 1988.
- [11] Kapur, K.C. and Lamberson, L.R., *Reliability in Engineering Design*, John Wiley and Sons, New York, 1977.

- [12] Klafter, Richard D., Chmielewski, T.A., and Negin, Michael, *Robotic Engineering: An Integrated Approach*, Prentice-Hall, New Jersey, 1989.
- [13] Kubat, P., "Assessing Reliability of Modular Software," *Operations Research Letters*, vol. 8, no. 1, pp. 35-14, February 1989.
- [14] Lee, K., Tillman, F., and Higgins, J., "A Literature Survey of the Human Reliability Component in a Man-Machine System," *IEEE Transactions on Reliability*, vol.37, no.1, pp. 24-34, April 1988.
- [15] Littlewood, B., "Software Reliability Model for Modular Program Structure," *IEEE Transactions on Reliability*, vol. 28, no. 3, pp. 241-246, August 1979.
- [16] Lin, Fen-Hui and Kuo, W., "Simulating Transient-State System Effectiveness for Human-Machine Systems," *IEEE Transactions on Reliability*, vol. 43, no. 4, pp. 569-574, December 1994.
- [17] Lydell, B., "Human Reliability Methodology: A Discussion of the State of the Art," *Reliability Engineering and System Safety*, vol. 36, pp. 15-21, 1992.
- [18] MIL-HDBK-217F, *Reliability Prediction of Electronic Equipment*, Department of Defense, Washington DC, 2 December 1991.
- [19] Moray, N., "Objective and Subjective Estimates of Human Error," *IEEE Transactions on Reliability*, vol. 38, no. 3, pp. 301-304, August 1989.
- [20] Musa, J.D., Iannino, A., and Okumoto, K., *Software Reliability: Measurement, Prediction, Application*, McGraw-Hill, New York, 1990.
- [21] O'Connor, Patrick D.T., *Practical Reliability Engineering*, 3rd ed, John Wiley and Sons, New York, 1991.
- [22] Paredis, C.J., Au, W.K. Frederick, and Khosla, P., "Kinematic Design of Fault Tolerant Manipulators," *Computers and Electrical Engineering*, vol. 20, no. 3, pp. 211-220, 1994.
- [23] Ramakumar, R., *Engineering Reliability: Fundamentals and Applications*, Prentice Hall, New Jersey, 1993.
- [24] Reliability Analysis Center, *Non-Electronic Parts Reliability Data (NPRD)*, NPRD-95, Rome Laboratory, Griffiss Air Force Base, NY, 1995.
- [25] *Reliability Engineer's Toolkit*, Rome Laboratory, Griffiss Air Force Base, NY, April 1993.

- [26] Repperger, D.W., and Koivo, A.J., "Characterization of Critical Human Tracking Regions and the Incidence of Pilot Induced Oscillation Using a Hidden Markov Process," Unpublished Report, Wright Patterson Air Force Base, OH, 1996.
- [27] Ross, Sheldon M., *Introduction to Probability Models*, 5th ed., Academic Press, London, 1993.
- [28] Schaaf, T.W. van der, "Human Recovery of Errors in Man-Machine Systems," *Analysis, Design, and Evaluation of Man-Machine Systems*, (Postprint Volume from the Sixth IFAC/IFIP/IFORS/IEA Symposium, p. 2, vol. xii+736), vol. 1, pp. 71-76, 1995.
- [29] Schneider, D., "Reliability and Maintainability of Modular Robot Systems: A Roadmap for Design," Dissertation, Mechanical Engineering Dept., University of Texas at Austin, Austin, TX, August 1993.
- [30] Seaver, D.A. and Stillwell, W.G., "Procedures for Using Expert Judgment to Estimate Human Error Probabilities in Nuclear Power Plant Operations," U.S. Nuclear Regulatory Commission, NUREG/CR-2743, Washington, D.C., 1983.
- [31] Shigley, J.E. and Mischke, C.R., *Mechanical Engineering Design*, 5th ed., McGraw-Hill, New York, 1989.
- [32] *SKF Bearings*, Product Catalog, Division of SKF USA Inc., Bethlehem, PA
- [33] Spong, M.W. and Vidyasagar, M., *Robot Dynamics and Control*, John Wiley and Sons, New York, 1989.
- [34] Swain, A.D., "Human Reliability Analysis: Need, Status, Trends and Limitations," *Reliability Engineering and System Safety*, vol. 29, pp. 301-313, 1990.
- [35] Wewerinke, P.H., Perdock, J., and van der Tak, C., "Model of the Human Observer and Controller of a Dynamic System - Theory and Model Application to Ship Handling," *Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, pp. 858-863, 1988.

Vita

Second Lieutenant Matthew J. Rummer was born on 15 July 1973 in Fort Monmouth, New Jersey. He graduated from Barberton High School in 1991 and entered undergraduate studies at Purdue University in West Lafayette, Indiana. He graduated with a Bachelor of Science degree in Mechanical Engineering in May 1995. He received his commission on 12 May 1995. In August 1995, he entered the School of Operations Research, Air Force Institute of Technology.

Permanent address: 1284 Benton St.
Barberton, OH 44203

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1997		3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE A Reliability Study of a Teleoperated Robotic System with Application to the Next Generation Munitions Handling System			5. FUNDING NUMBERS	
6. AUTHOR(S) Matthew J. Rummer, Second Lieutenant, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology/ENY 2950 P Street Wright-Patterson AFB, Ohio 45433			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GOR/ENY/97M-02	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Munition Material Handling Equipment (MMHE) Focal Point AFSEO/SKZ 207 W. D Avenue, Suite 303 Eglin AFB, FL 32542-6845			10. SPONSORING/MONITORING AGENCY REPORT NUMBER N/A	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE a	
13. ABSTRACT (Maximum 200 words) The United States Air Force is currently developing an Advanced Technology Demonstrator (ATD) for the Next Generation Munitions Handling System (NGMH). The NGMH/ATD system is being designed at Oak Ridge National Laboratory as a teleoperated robotic manipulator. In this research, a reliability model for a general telerobotic manipulator is developed including: (1) human, (2) software, (3) hardware, and (4) kinematic components. Each of these components contributes towards the reliability of the system, where reliability is defined as the probability of the end-effector of the manipulator being within a certain error bound. The model provides a comparative measure of reliability based on a combination of two underlying models. A semi-Markov model is used for the human, software, and hardware components, while the kinematic reliability estimates are obtained through a separate simulation. The reliability model developed is not intended to be a predictive model, rather to provide a means of comparing different configurations of a telerobotic system.				
14. SUBJECT TERMS Reliability, Teleoperated Robotic System, Teleoperators, Robotics, Manipulators, Kinematic Reliability, Telerobotic Manipulator, Munitions Handling			15. NUMBER OF PAGES 140	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to **stay within the lines** to meet **optical scanning requirements**.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."

DOE - See authorities.

NASA - See Handbook NHB 2200.2.

NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.

DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

NASA - Leave blank.

NTIS - Leave blank.

Block 13. Abstract. Include a brief (*Maximum 200 words*) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (*NTIS only*).

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.